St. Cloud State University theRepository at St. Cloud State

Culminating Projects in Information Assurance

Department of Information Systems

5-2019

Automated Teller Machine Ethernet Traffic Identification to Target Forensics Detection of IP Packets

Brian Volkmuth bgvolkmuth@stcloudstate.edu

Follow this and additional works at: https://repository.stcloudstate.edu/msia_etds

Recommended Citation

Volkmuth, Brian, "Automated Teller Machine Ethernet Traffic Identification to Target Forensics Detection of IP Packets" (2019). *Culminating Projects in Information Assurance*. 87. https://repository.stcloudstate.edu/msia_etds/87

This Starred Paper is brought to you for free and open access by the Department of Information Systems at the Repository at St. Cloud State. It has been accepted for inclusion in Culminating Projects in Information Assurance by an authorized administrator of the Repository at St. Cloud State. For more information, please contact rswexelbaum@stcloudstate.edu.

Automated Teller Machine Ethernet Traffic Identification to Target Forensics Detection of IP Packets

by

Brian Volkmuth

A Starred Paper

Submitted to the Graduate Faculty

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in Information Assurance

April, 2019

Starred Paper Committee: Abdullah Abu Hussein, Chairperson Lynn A. Collen Balasubramanian Kasi

Abstract

Over the last few decades, consumers have become accustom to the convenience of Automatic Teller Machines (ATMs) to transfer funds between accounts, provide account balance information and to withdraw cash from savings, checking, and other account types. Along with the convenience and ease of locating an ATM through mobile bank apps, there has been a significant increase in ATM fraud across the globe. Consumer confidence in the ATM, bank and credit card issuer is greatly impacted by the perceived level of security in ATM transactions and the technology behind them. Confronting the risk associated with ATM fraud and limiting its impact is an important issue that face financial institutions as the sophistication of fraud techniques have advanced. Largely the process behind the verification of these transactions has moved from Plain Old Telephone System (POTS) to Ethernet connections to the processors, banks and card issuers. The attack surface has grown, both in size and complexity. These security risks should be prompting the industry to research all attack surfaces, and this research looks specifically the Ethernet packets that make up these types of transactions. In this research, I investigate the packet structure and predictability within ATM Ethernet traffic. Even with the proliferation of retail ATMs in the most common of retail spaces, this attack vector has received little attention.

Acknowledgements

I would like to thank Dr. Abdullah Abu Hussein for his patient help, guidance, and encouragement and Dr. Collen Lynn and Dr. Balasubramanian Kasi for their thoughtful participation on the committee and excellent feedback and of course my wife, Alicia Volkmuth MBA, MA.

Table of Contents

	Page
List of	Tables6
List of	Figures7
Chapt	er
I.	Introduction
	Introduction8
	Problem Statement9
	Nature and Significance of the Problem 12
	Objective of the Study 14
	Study Questions/Hypotheses
	Limitations of the Study15
	Definition of Terms
II.	Background and Review of Literature25
	Introduction
	Background Related to the Problem26
	Literature Related to the Problem27
	Literature Related to the Methodology 30
III.	Methodology
	Introduction
	Design of the Study
	Data Collection

Chapt	er	Page
	Tools and Techniques	
	Hardware and Software Environment	
IV.	Data Presentation and Analysis	35
	Introduction	
	Data Presentation	35
	Data Analysis	43
V.	Results, Conclusion, and Recommendations	45
	Introduction	45
	Results	45
	Conclusion	
	Future Work	53
Refere	ences	55
Appen	ndices	
A.	Successful Withdrawal \$20	59
B.	Successful Withdrawal \$60	61
C.	Balance Check Successful Zero Balance	63
D.	Insufficient Funds	65
E.	Balance Check, Wrong PIN	67
F.	Successful Balance Check \$110	69
G.	Transaction Data	71
Н.	Packet Sequence Comparison by Packet Number	72

List of Tables

Table		Page
1.	Packets Captured for Five Test Transactions	39
2.	Packets Captured for Five Test Transactions (2019)	40
3.	Data Packets with Overhead Packets Removed	42
4.	Typical Transaction Packet Data and Information	43
5.	Final Packet for Analysis	46
6.	ATM Start-up Capture	51

List of Figures

Figure		Page
1.	ATM Cloud Transactions	10
2.	ATM Transaction Flow Chart with Potential Traffic Intercept Points	11
3.	Common ATM Network Diagram	33
4.	Congested Production Network Packet Capture Topology	334
5.	ATM Start-up Authentication Packet Capture	
6.	Initial ATM Authentication Packet Capture	37
7.	Filtered Packet Capture	38
8.	Overhead Packet	41

Chapter I: Introduction

Introduction

As with many conveniences of the modern age, the ATM has brought with its popularity the chance of compromised personal data. "Fraud perpetrated at the Automated Teller Machine (ATM) has suddenly and somewhat unexpectedly exploded to the highest rate in two decades" (Sidel, 2015). The uptick in fraud cases is largely due to the new more secure credit and debit cards that banks and credit card companies are rushing to replace the 30-year-old magnetic strip technology. "Criminals 'know there is still vulnerability [at the ATM] and they are trying to capitalize on it,' said Owen Wild, director of security marketing at NCR Corp., on the of the largest ATM manufactures" (Sidel, 2015). While the majority of these cases involve capturing the information at the point of reading the card, this research is looking at the possible vulnerability from and on the network that the ATM is connected, the construction of the packets, and the security protocols used in the TCP portion of the packets..

ATMs (or Automated Teller Machines) are what are known as *Alternative Delivery Channels*.

- *Alternative*, because they are alternative to the human teller.
- Delivery because they are programmed to deliver a specific set of services.
- *Channel* because in the banking world, all delivery mechanisms need a *channel* via which the delivery would be made (Khan, 2015).

The security of ATM transactions requires a continuous and proactive review of physical components of the machine related to its connection to a network, the security in the protocols used, and operating system security. The challenges of the market include :

- Constant evolution of threats
- 30-year-old technology still in use in the form of magnetic stripe cards
- Transition to new embedded chip cards
- Prevalent use of legacy operating systems

Analyzing ATM originated Ethernet traffic to create a template for forensics analysis of packets in order to provide a signature for firewalls, determine vulnerabilities based on the characteristics of the traffic, and what follow on work could be recommended, is the goal of this research. This research will use a common off-theshelf protocol analyzer and determine if a forensics investigator could identify ATM traffic using a packet inspection profile without the knowledge of the sending or receiving IP information. Should a benchmark be possible from this work, there are many other avenues to explore in the retail ATM market.

Problem Statement

We want all of our ATM transactions to be safe and secure even while the transactional data is in transit from the ATM over the internet to the processing servers and back.

Today, more and more ATMs are connecting to their processing servers via the Internet with Ethernet and, in some cases, Wi-Fi connections. There are significant vulnerabilities that need to be explored to create better-protected transactions for consumers. Captured transactional data packets by the use of a packet sniffer could allow the capture and nefarious use of private information and harm consumers, retailers and the banking industry. We will use packet sniffing software to capture packets from different types of ATM transactions, evaluate them for predictability, and determine the level of ease in finding transactions in a congested production network.

All ATMs connect to a server to process transactions. As seen in Figure 1, the ATM network connects to the cloud as do all of the entities involved in a transaction. The internal bank network is connected to the cloud and pushes account information updates to an ATM Outsource Network Provider. When an ATM or Point of Sale (POS) machine initiates a transaction, the ATM Outsource Network Provider handles the verification of the card and PIN if used, that sufficient funds are available for the requested transaction, among other details. In the past, all ATMs completed transactions via POTS. Today, there are a number of communication technologies being employed including cabled Ethernet, Wireless Ethernet, cellular data, and satellite internet connections. Each of these presents a unique attack surface, but all depend on an Ethernet connection at some point between the ATM and processing server.



Figure 1. ATM Cloud Transactions

When a transaction is initiated, it follows a flow chart approximated in Figure 2. One of the parts of this study is to determine if it is possible, as the customer works through the decision tree, to see when data is transmitted. In the previous work, data was only transmitted once all the prompts had been satisfied and the customer chose what amount of money was to be withdrawn and from which account. With the advent of EMV technology which included software and hardware upgrade, this may no longer be true.



Figure 2. ATM Transaction Flow Chart with Potential Traffic Intercept Points

Nature and Significance of the Problem

The ATM industry has had several government and self-mandated software and equipment updates in a very short period of time. They include changes to the American with Disabilities Act (ADA) elements of an ATM; industry changes to the new and much more secure Euro pay, MasterCard, and Visa (EMV) card technology; and an industry change from dial-up modems to Ethernet and Internet Protocol (IP) based communications. The sheer volume of changes has created a confusing and challenging landscape of regulatory compliance, security gaps, and cost. Combined, this environment has attracted hackers and organized crime groups to pounce on the security gaps and is a world-wide issue. The adaption of current technologies has prompted a change from transmitting transaction data over phone lines to Ethernet connections to the cloud with an encrypted payload.

Recently, due to heavier computing demands and the falling price of computerlike architectures, ATMs have moved away from custom hardware architectures using microcontrollers and / or application-specific integrated circuits, most of which are based on Intel 8086 architecture, to adopting a hardware architecture that is very similar to a personal computer. Many ATMs are now able to use commercial operating systems such as Microsoft Windows and Linux. Although it is undoubtedly cheaper to use commercial off–the–shelf hardware, it does make ATMs vulnerable to the same sort of problems exhibited by conventional computers. (Adeoye, 2012, p. 457) Logical attacks on ATMs are on the rise. ATM fraud has increased in sophistication and organization. These organizations have become adept in developing methods and hardware to take advantage of new technologies such as miniaturization, battery life and communications methods like WiFi, Bluetooth, and Near Field Communication (NFC).

Remote attacks target the ATM networks and attempt to compromise the communication with the host. These attacks are more critical because a hacker does not need to open up the ATMs. As ATM technology knowledge becomes widespread, monitoring systems access through web browsers or TELNET enables an easy access to attackers who can hijack ATM management systems and perform management functions. ATM networks are still vulnerable to similar IP based network attacks. Remote attacks such as Eavesdropping, Spoofing, Denial of Service, Sniffing and Virtual Channel Theft are almost always carried out by criminal organizations. (Chafalon 2012, ¶11-13)

Souvignet, Hatin, Maqua, Tesniere, Léger, and Hormière (2014) reported that there are two main classifications of payment card fraud, card not present and physical fraud. The one we are concerned with here is "physical fraud, where data originates from lost and stolen cards, skimming, shimming, man in the middle attacks, Automated Teller Machine (ATM) reverse engineering, etc." (p. 143).

Along with the hardware and software changes, there has been a significant shift from dial-up to Ethernet and wireless communication via the internet. This is the focus of this research, specifically looking at packet dumps and identifying ATM specific traffic based on a multi-point inspection method. It is the norm when doing network forensics or traffic analysis to look at port numbers. In this case the port number is standard, 443, and will aid identification of the target traffic but it will not be the sole metric used. It should be noted that "the traditional approach that relies on transport layer port numbers is largely limited with less than 70% accuracy for application identification" (Alblawi & Jinoh, 2014, p. 4). With that in mind, the decision was made to include a study of the encrypted payload and header information with the exclusion of the destination IP address. To deal with the limitations of port-based inspection, payload information or statistical flow features were used. The payload-based identification inspects application data for traffic classification such as search application-specific signatures. Some interesting work has been done in this area with very good results.

A machine learning based approach employing simple packet header feature sets and statistical flow feature sets without using the IP addresses, source/destination ports and payload information to unveil encrypted application tunnels in network traffic. We demonstrate the effectiveness of our approach as a forensic analysis tool on two encrypted applications Secure Shell (SSH) and Skype, using traces captured from entirely different networks. (Alshammari & Zincir-Heywood, 2010, p. 1326)

Objective of the Study

The objective of this study is to formulate an initial set of network forensics metrics that determine the predictability of current Automated Teller Machine Ethernet packets transmitted initially on local area networks and eventually over the internet. The study should determine the number of packets generated during different types of transactions, packet size, packet order and determine what human readable and encrypted components exist in each packet type.

Study Questions/Hypotheses

- 1. How many packets make up each of the six transactions types?
 - a. ATM Startup
 - b. Insufficient Funds
 - c. Balance Check Incorrect PIN
 - d. Balance Check Successful
 - e. \$20 Withdrawn
 - f. \$60 Withdrawn
- 2. Is each packet of predictable length?
- 3. Is packet length influenced by transaction type?
- 4. What protocols are used throughout each transaction?
- 5. What known vulnerabilities exist regarding the packet construction and protocols used in ATM transaction packets?

Limitations of the Study

This study is limited to the developing an understanding of the number, size and repeatable patterns that may occur within several different ATM transaction types. Due to legal limitations, we are unable to do any analysis of the packet payloads and the cryptography contained within them.

Definition of Terms

3DES("triple DES"). Triple DES applies the Data Encryption Standard (DES) cipher algorithm three times to each data block. In general, TDES was introduced to have three keys. Having a key length of 168 bits: three 56-bit DES keys. When it was discovered that a 56-bit key of DES is not enough to protect from attacks, TDES was chosen as a simple way to enlarge the keyspace without a need to switch to a new algorithm. Triple DES simply extends the key size of DES by applying the algorithm three times in succession with three different keys (Debnath, Agrawal, & Vaishnav 2014).

ATM Vault. The ATM Vault is a secure section of an ATM, separate from the bezel containing the card reader, screen, EPP Pin Pad and receipt printer. It contains the Cash dispensing unit and is secured by either an electronic or dial combination lock.

Americans with Disabilities Act of 1990 (ADA). ADA is a civil rights law that prohibits discrimination against individuals with disabilities in all areas of public life, including jobs, schools, transportation, and all public and private places that are open to the general public. The purpose of the law is to make sure that people with disabilities have the same rights and opportunities as everyone else (ADA National Network).

Automated Teller Machine (ATM). An ATM is a computerized electronic machine that performs basic banking functions (such as handling check deposits or issuing cash withdrawals) (Merriam-Webster's collegiate dictionary 2018).

Bluetooth. A standard for the short-range wireless interconnection of mobile phones, computers, and other electronic devices (Bluetooth, n.d.).

Card Holder. A person who has a credit or debit card.

Card Issuer. A bank or credit union who offers credit cards. When consumers make credit card purchases, the credit card issuer is responsible for sending payments to merchants for purchases made with credit cards from that bank.

Cash Dispensing Unit (CDU). Within the vault of an ATM is the cash dispensing unit. The CDU has electronic sensors that count each bill as it exits the dispenser. It also has sensors that detect damaged bills, doubles or bills stuck together and that detect jammed bills.

Comma Separated Value (CSV). A simple file format used to store tabular data, such as a spreadsheet or database. Files in the CSV file format can be imported to and exported from programs that store data in tables such as Microsoft Excel or OpenOffice Calc.

EMV Card Reader. An electronic sensor that reads an EMV chip, magnetic strip, bar code on a credit card, membership card, etc. (ATM of America, n.d.).

EMV Chip (Euro pay, MasterCard, and Visa). A payment instrument technology with an embedded microprocessor chip that store and protect cardholder data (ATM of America, n.d.).

Encrypting PIN Pad (EPPs). A form a component of unattended PIN Entry Devices. Typically, EPPs are used to enter a cardholder's PIN in a secure manner. EPPs are used in conjunction with ATMs, automated fuel dispensers, kiosks, and vending machines. Requirements for these devices are found in security documents published by the Payment Card Industry (ATM of America, n.d.). **Ethernet**. The traditional technology for connecting wired local area networks (LANs), enabling devices to communicate with each other via a protocol -- a set of rules or common network language. As a data-link layer protocol in the TCP/IP stack, Ethernet describes how network devices can format and transmit data packets so other devices on the same local or campus area network segment can recognize, receive and process them. An Ethernet cable is the physical, encased wiring over which the data travels (TechTarget Network, n.d.a).

Ethernet Hub. A basic type of wired network device that allows multiple connected computers to communicate via broadcast communication.

Firewall. Part of a computer system or network that is designed to block unauthorized access while permitting outward-bound communication (Firewall, n.d.).

Frame. A digital data transmission unit in computer networking and telecommunication. A frame typically includes synchronization features consisting of a sequence of bits or symbols that indicate to the receiver, the beginning, and end of the payload data within the stream of symbols or bits it receives (Cisco Networking Academy, 2017).

Input/output Board (I/O Board). A circuit board that contains physical connections to peripheral devices. In an ATM these include the receipt printer, card reader, and cash dispensing unit, LCD screen, and EPP Pin Pad (ATM of America, n.d.).

Internet Protocol (IP). The principal communications protocol in the internet protocol suite for relaying datagrams across network boundaries. Its routing function

enables internetworking and essentially establishes the internet (Cisco Networking Academy, 2017).

IP Packet. In computer networks, a pack is a container or box that carries data over a TCP/IP network and internetworks. A packet contains several pieces of information, including the data it is carrying, source and destination IP addresses and other constraints required for quality of service and packet handling (Cisco Networking Academy, 2017).

Local Area Network (LAN). A local area network (LAN) is a group of computers and associated devices that share a common communications line or wireless link to a server. Typically, a LAN encompasses computers and peripherals connected to a server within a distinct geographic area such as an office or a commercial establishment. Computers and other mobile devices use a LAN connection to share resources such as a printer or network storage (TechTarget Network, n.d.b).

Logical Port. In programming, a port is a "logical connection place" and specifically, using the Internet's protocol, TCP/IP, the way a client program specifies a particular server program on a computer in a network. Higher-level applications that use TCP/IP such as the Web protocol, Hypertext Transfer Protocol, have ports with preassigned numbers (Cisco Networking Academy, 2017).

Main Board. Sometimes knows as the motherboard, this is the main printed circuit board found in general purpose microcomputers such as in an ATM and other expandable systems. It holds and allows communication between many of the crucial

electronic components of a system. Such as the central processing unit (CPU) and memory and provides connectors for other peripherals (ATM of America, n.d.).

Near Field Communication (NFC). A technology allowing the short-range wireless intercommunication of mobile phones and other electronic devices for purposes such as making payments.

Network Switch. A computer networking device that connects devices together on a computer network by using packet switching to receive, process, and forward data to the destination device. A network switch is a multi-port network bridge that uses hardware addresses to process and forward data at the data link layer (layer 2) of the OSI model. Some switches can also process data at the network layer (layer 3) by additionally incorporating routing functionality. Such switches are commonly known as layer-3 switches or multilayer switches (Cisco Networking Academy, 2017).

Open Systems Interconnect (OSI) Model. A conceptual model that characterizes and standardized the communication functions of a telecommunication or computing system without regard to its underlying internal structure and technology.

- Layer 1: Physical Layer. Defines electrical and physical specifications for devices. The physical layer defines the relationship between a device and a transmission medium, such as a copper or optical cable.
- Layer 2: Data Link Layer. Provides the functional and procedural means to transfer data between network entities and to detect and possibly correct errors that may occur in the physical layer.

20

- Layer 3: Network Layer. Provides the functional and procedural means of transferring variable length data sequences from a source host on one network to a destination host on a different network, while maintaining the quality of service requested by the transport layer (in contrast to the data link layer which connects hosts within the same network).
- Layer 4: Transport Layer. Provides transparent transfer of data between end users, providing reliable data transfer services to the upper layers of the OSI model. The transport layer controls the reliability of a given link through flow control, segmentation and desegmentation, and error control (Cisco 2017)

Packet Sniffer. A program that can record all network packets that travel past a given network interface, on a given computer, on a network. It can be used to troubleshoot network problems, as well as to extract sensitive information such as credentials from unencrypted login sessions.

Payment Card Industry (PCI). The segment of the financial industry that governs the use of all electronic forms of payment.

Payment Card Industry Security Standards Council (PCI SSC). This oversees policies and technologies behind non-cash payments including transactions involving credit cards, prepaid cards, point-of-sale cards, e-purse, bank debit, and ATM cards.

Payment Processor. A company (often a third party) appointed by a merchant to handle transactions from various channels such as credit cards and debit cards for merchant acquiring banks. They are usually broken down into two types: front-end and back-end.

Personal Identification Number (PIN). An identifying number allocated to an individual by a bank or other organization and used for validating electronic transactions.

Physical Port. On computer and telecommunication devices, a port is generally a specific place for being physically connected to some other device, usually with a socket and plug (Cisco Networking Academy, 2017).

Plain Old Telephone Service (POTS). A retronym for voice-grade telephone service employing analog signal transmission over copper loops. POTS was the standard service offering from telephone companies from 1876 until 1988. POTS remains the basic form of residential and small business service connection to the telephone network in many parts of the world. The term reflects the technology that has been available since the introduction of the public telephone system in the late 19th century, in a form mostly unchanged despite the introduction of Touch-Tone dialing, electronic telephone exchanges and fiber-optic communication into the public switched telephone network (Plain old telephone service, n.d.).

Router. A networking device that forwards data packets between computer networks. Routers perform the traffic directing functions on the Internet. Data sent through the internet, such as a web page or email, is in the form of data packets. A packet is typically forwarded from one router to another router through the networks that constitute an internetwork until it reaches its destination node. A router is connected to two or more data lines from different networks. When a data packet comes in on one of the lines, the router reads the network address information in the packet to determine the ultimate destination. Then, using information in its routing table or routing policy, it directs the packet to the next network on its journey (Cisco Networking Academy, 2017).

Secure Hash Algorithm 1 (SHA-1). A cryptographic hash function which takes an input and produces a 160-bit (20-byte) hash value known as a message digest, typically rendered as a hexadecimal number, 40 digits long.

Secure Hash Algorithm SHA-256. One of a number of cryptographic hash functions. SHA-256 algorithm generates an almost unique, fixed size 256-bit (32-byte) has.

Secure Socket Layer (SSL). A standard security technology for establishing an encrypted link between a web server and a browser. This link ensures that all data passed between the web server and browsers remain private and integral. This protocol has now been depreciated by the Internet Engineering Task Force (IETF) and has largely been replaced by Transport Layer Security (Cisco Networking Academy, 2017).

TLSv1.2. The newest SSL protocol version. It introduces new SSL/TLS cipher suites that use the SHA-256 hash algorithm instead of the SHA-1 function, which adds significant strength to the data integrity.

Telnet. A network protocol that allows a user on one computer to log onto another computer that is part of the same or remote network (Cisco Networking Academy, 2017).

Terminal Number. A unique alpha-numeric identifier for each ATM terminal that identifies it to the processing system and is used in combination with the 3DES validation process to make the initial authentication of the terminal to the processing system. It is also used to identify the specific machine during each transaction processed by that terminal (ATM of America, n.d.).

Transport Layer Security (TLS). A protocol that provides privacy and data integrity between two communicating applications (Cisco Networking Academy, 2017).

Uniform Resource Locator (URL). The address of a resource on the Internet. A URL indicates the location of a resource as well as the protocol used to access the resource (Cisco Networking Academy, 2017).

Wi-Fi. The technology for radio wireless local area networking of devices based on the International Electronic and Electrical Engineers (IEEE) 802.11 standards (Cisco Networking Academy, 2017).

Chapter II: Background and Review of Literature

Introduction

There are two areas of importance to this proposal. First, successfully capturing packet data from a variety of ATM transaction types. And second, the determination packet data predictability. In previous work, both aspects were proven to be correct and repeatable. However, this may not be the case with the large number of security and technology changes made in the intervening four years. The specific changes are:

- Updated and patched Windows CE 6.0
- Updated firmware for:
 - o EMV Card Reader
 - Cash Dispensing Unit
 - PCI Compliant Key Pad
 - Receipt Printer
- Hardware and software upgrade from a magnetic stripe card reader to EMV chip reader.
 - o New card reader
 - o New software
 - New internal cables and grounding connections
- Card being swiped to card being dipped and held during the transaction.
- Network settings for FIS changed from a public IP address to a URL.
- Socket layer security changed from SSL to TLSv1.2 or higher.

- FIS server settings changed to reject connection attempts that do not meet the new minimum settings for EMV upgrades.
 - o Must use URL
 - o TLSv1.2 or higher
 - EMV must be configured on the terminal with appropriate PIDs activated
 - Software version 6.1.19 or higher must be installed.

In the course of this literature review, known ATM attacks and vulnerabilities, as well as the packet sniffing process using common Windows-based open source tools will be explored.

Background Related to the Problem

There has been a significant shift from POTS to Ethernet for secure connections between ATMs and their processing servers. Along with this shift, ATMs have become almost ubiquitous in all retail settings. The manufacturer of the ATM used in this study reports sales in excess of 100,000 units in the United States alone. According to Pymnts.com (2017), the "number of ATMs in the U.S. now stands between 475,000 and 500,000 in 2017" (¶ 1). They are found in convenience stores, bars, restaurants, hotels/motels, and even at short-lived events such as outdoor music festivals and fairs. In some cases, the businesses involved do not own the ATM but do provide the communication infrastructure and in the case of Ethernet, lack the expertise to secure their networks, nor understand what a compromised network looks like. In many locations, access to the network infrastructure is both easy and not monitored. Using a

Packet Sniffer, a compact host device and basic knowledge, the transactions that make up the startup communication between the ATM and processing server and transactions themselves.

Literature Related to the Problem

There is very little to be found in literature related specifically to packet capture of ATM traffic. The vast majority discuss security as it relates to customer satisfaction, physical security, skimmers, malware, and other much more common threats. Therefore, literature has been included literature where security threats from the network are mentioned but rarely described beyond that they are threats found on a network.

The ATM has become as, if not more, ubiquitous than pay telephones were in the past. They can be found in nearly every local eating establishment, convenience store, airport, college, and even hospitals. There are a variety of methods for the ATM to connect to its processing and validating server which include POTS, Ethernet, WiFi, and direct connection to a computer host as a peripheral device.

The benefits of ATM that can be derived from ATM usage are so numerous, some are outlined below:

- Flexible account access allows clients to access their accounts at their convenience.
- MFI (bank) personnel are not required to be present for transactions and have more time to serve clients.
- Increased hours of operation fit client schedules.

- More clients can be reached beyond the branch network, such as in smaller population centers.
- More low-cost funds are available because ATMs make it easier for clients to deposit savings. (Adepju & Alhassan, 2010, p. 4-5)

The security threats are described in how they affect the convenience of ATMs as a service but often not in terms of threats to the user's private data. The convenience that ATMs provide comes with inherent risks both to the consumer as well as the ATM owner.

ATM fraud is not the sole problem of banks alone; it is a big threat, and it requires coordinated and cooperative action on the part of the bank, customers, and the law enforcement machinery. The ATM frauds not only cause financial loss to banks, but they also undermine customers' confidence in the use of ATMs. (Brunner, Decressin, Hardy, & Kudela, 2004)

The threats to ATMs are both local to the individual machine and can be found in the cloud services that provide authentication and transactional information between the ATM, cardholder and processing agent's infrastructure. "With the introduction of internet technology in recent years, the Internet communication is exposed to unwanted people giving them access to pose different kinds of attacks on ATM Systems" (Twum, Nti, & Assante, 2016, p. 126). Within this statement, the importance of this study can be found and supported. With the significant shift from POTS as the primary means of communication to Ethernet-based technologies, opens up the means to capture transaction data and allow criminals to raid both the ATM, customer's accounts and consumers' confidence in bank and credit card companies. The gains by criminals have been massive in scope and lighting quick in implementation.

In 2013, Ghana Commercial Bank (GCB) confirms money theft from an ATM of about GH¢3 million (Obour 2013) and a worldwide gang of criminals stole \$45 million in a matter of hours by hacking their way into a database of prepaid debit cards and then draining cash machines around the globe. (Modern Ghana, 2013, p. 127)

The scope of an effect on consumer confidence is an often-studied phenomenon in many countries throughout the world. This provides the motivation to assess the security risks, develop standards and create benchmarks to more efficiently and definitively discover and evaluate compromised aspects of an ATM transaction. "Since it is a day of technology, innovation, and cyber-hacking, so security and privacy of customer's information (Pin code, password, etc.) is a burning issue that has an impact on customer's satisfaction and dissatisfaction" (Hossain, Russel, & Robidas, 2015, p. 68). Customer satisfaction impacts due to security concerns directly relate to this study. While the vast infrastructure comprising the ATM and mobile banking system is largely out of sight to the majority of consumers, the attack surface, even if only looking at the network related portion is large, has many intrusion points and each is likely controlled by an entity with little or no relationship to another nor have the expertise to adequately physically or logically secure their network infrastructure. For many businesses, with regards to ATMs, they relied on POTS communication connections in the past. This required very little in the way of physical or logical security. Ethernet security is well beyond what they have experienced and often is not a priority.

Literature Related to the Methodology

Capturing and determining packet type, length, protocols used, encryption found if any, and addressing information are the significant data types being sought with this work. To that end, a packet sniffer configured in promiscuous mode will be used.

IP based sniffing is the most commonly used method of packet sniffing. In this method, a requirement of setting network card into promiscuous mode exist. When network card is set into promiscuous mode, then the host will be able to sniff all packets. (Rupam, Verma, & Singh, 2013, p. 23)

The software Wireshark is a powerful network or protocol analyzer, otherwise known as a packet sniffer. It can be used to look in detail at the structure and content of packets captured from the network interface on the host computer. The features of Wireshark include:

- Data is analyzed either from the wire over the network connection or from data files that have already captured data packets.
- Supports live data reading and analysis for a wide range of networks (including Ethernet, IEEE 802.11, point-to-point Protocol (PPP) and loopback).
- With the help of GUI or other versions, users can browse captured data networks.

- For programmatically editing and converting the captured files to the editcap application, users can use command line switches.
- Display filters are used to filter and organize the data display.
- New protocols can be scrutinized by creating plug-ins. (Wireshark, n.d.)

The end goal of this work is to determine if a benchmark of predictable traffic can be found. In the end, this could be useful in further use of a packet sniffer to monitor network assets.

The most common goal of packet sniffing is to monitor network assets to detect anomalous behavior and misuse. This concept has been around for nearly twenty years, but only recently has it seen a dramatic rise in popularity and incorporation into the overall information security infrastructure. (Banerjee, Vashishtha, & Saxena, 2010, p. 2)

Due to the dearth of literature on the use of a packet sniffer to specifically capture ATM traffic, the methodology of packet capture is generic in nature at the proposed level of granularity for this study.

Chapter III: Methodology

Introduction

This section outlines creating the test environment used for ATM transactions over Ethernet networks with the use of a packet analyzer. Owing to the limited availability of a real-world e-commerce Ethernet network to perform this experiment, the ATM will be connected to a network in an industrial setting to simulate a congested retail network for this study. We will discuss the setup and design of the network and the structured approaches that will assist in this study.

Design of the Study

To answer the given research question, quantitative methods would be most appropriate to find an answer. For the data samples collected commonly found in the market place retail ATM, a debit card, and running six transaction types:

- Successful Withdrawal \$20
- Successful Withdrawal \$60
- Successful Balance Check Zero Balance
- Successful Balance Check \$110
- Insufficient Funds
- Failed Transaction Expired Card

Data Collection

The initial set of data will be collected in small network will little overhead or other traffic to obtain an understanding of each of the transmissions more easily. As seen in Figure 3, the ATM is connected to a switch which is then connected to a hub, then to a

router and out to the cloud. The packet sniffing host is connected to the hub to provide access to all communication on the LAN.



Figure 3. Common ATM Network Diagram

After the first data set has been analyzed and any repeatable metrics have been determined, the second set of data will be collected in a much more congested production network. The ATM will be placed deep within a segmented network to create a condition in which a packet sniffer will have to contend with a much larger number of packets. This could create the possibility of packets with similar characteristics of those found in ATM packets in the first data set.



Figure 4. Congested Production Network Packet Capture Topology

Tools and Techniques

To analyze the packet data collected in WireShark, the following information was collected for each transaction and evaluated against each transaction for the following metrics:

- The initial ATM machine start up and connect to the processing server
- Number of packets per transaction
- Size, in bytes, found in each packet

Hardware and Software Environment

This study required the use of a Windows workstation, running Windows 10, Wireshark Network Analyzer v2.6.4, a hub, a border network segment, production network, a retail ATM with the latest OS, software and firmware patches, and a credit card.

Chapter IV: Data Presentation and Analysis

Introduction

The data collected in the research was collected as designed without complications. The process of mining the data for the elements related to this research was also completed without issue. The very capable filters available in WireShark made this process efficient and allowed the results to be saved as a Comma Separated Value (CSV) file, opened in Microsoft Excel and further sorted and analyzed. The results, however, show some expected and unexpected elements that require some manipulating of the data to get to the sought-after information. The process of culling the data is described for each of the research questions as needed.

Data Presentation

Unlike the original research project, the data stream between the ATM and the processing server at startup was captured in order to determine if the new requirements relating to ENV chips, changing to TLSv1.2, and updated and patched OS and PAI software. At the time of the previous work, a more basic authentication process to the server that occurs each time the ATM is powered up, often used POTS versus Ethernet which has since become more prevalent. In addition to a large shift to using Ethernet instead of POTS, the authentication process has grown in complexity and duration. The addition of EMV card readers, TLSv1.2, and the latest PCI compliant keypads are the cause of the new complexity while the OS and PAI software and peripheral firmware versions remain a component of the startup authentication checks but did not add anything new. The new changes created the curiosity of what that looks like from the
perspective of the number, size, and protocols found in the data packets, led to the first data capture. This data set was also used this to verify that the network segment used had a considerable amount of traffic. In Figure 5 below, a truncated display of the first WireShark capture of approximately 2 minutes and 36 seconds during which nearly 73,000 packets were captured.

	ATM_Startup.pcapng							
Ei	<u>File Edit Yiew Go Capture Analyze Statistics Telephony Wireless Tools H</u> elp							
	🚄 🔳 🖉 🐵 📕 🛅 🛛 🗳 🗮 🍝 🗮 🔜 🍳 🍳 🎕 🎹							
	Apply a display filter <ctrl-></ctrl->							
No	o. Time	Source	Destination	Protocol	Length Info			
F	1 0.0000	13.107.4.50	192.168.0.5	HTTP	1514 Continuation			
	2 0.0000	192.168.0.5	13.107.4.50	TCP	54 56690 → 80 [ACK] Seq=1 Ack=1461 Win=1006 Len=0			
	3 0.0012	13.107.4.50	192.168.0.5	HTTP	1514 Continuation			
	4 0.0025	13.107.4.50	192.168.0.5	HTTP	1514 Continuation			
	5 0.0026	192.168.0.5	13.107.4.50	TCP	54 56690 → 80 [ACK] Seq=1 Ack=4381 Win=1006 Len=0			
	6 0.0037	13.107.4.50	192.168.0.5	HTTP	1514 Continuation			
	7 0.0049	13.107.4.50	192.168.0.5	HTTP	1514 Continuation			
	8 0.0050	192.168.0.5	13.107.4.50	TCP	54 56690 → 80 [ACK] Seq=1 Ack=7301 Win=1006 Len=0			
	9 0.0061	13.107.4.50	192.168.0.5	HTTP	1514 Continuation			
	10 0.0074	13.107.4.50	192.168.0.5	HTTP	1514 Continuation			
	11 0.0075	192.168.0.5	13.107.4.50	TCP	54 56690 → 80 [ACK] Seq=1 Ack=10221 Win=1006 Len=0			
	12 0.0086	13.107.4.50	192.168.0.5	HTTP	1514 Continuation			
	13 0.0098	13.107.4.50	192.168.0.5	нттр	1514 Continuation			
	14 0.0099	192.168.0.5	13.107.4.50	ТСР	54 56690 → 80 [ACK] Seq=1 Ack=13141 Win=1006 Len=0			
	15 0.0111	13.107.4.50	192.168.0.5	HTTP	1514 Continuation			
h	16 0.0123	13.107.4.50	192.168.0.5	HITTO				
	72025 142 04	102 168 0 5	12 107 4 50	TCD	54 56600 . 00 [ACK] Seq=11327 Ack=1751020 Up=1006			
	72925 142.84	192.108.0.5	13.107.4.50	TCP	54 50090 → 80 [ACK] SEQ=111327 ACK=17519229 WIN=1000			
	/2926 142.84	13.107.4.50	192.168.0.5	TCP	1514 80 → 56690 [ACK] Seq=1/519229 ACK=11132/ W1n=1026			
	/292/ 142.85	13.107.4.50	192.168.0.5	ТСР	1514 80 → 56690 [ACK] Seq=1/520689 Ack=11132/ Win=1026			
	72928 142.85	192.168.0.5	13.107.4.50	TCP	54 56690 → 80 [ACK] Seq=111327 Ack=17522149 Win=1006			
	72929 142.85	13.107.4.50	192.168.0.5	TCP	1514 80 → 56690 [ACK] Seq=17522149 Ack=111327 Win=1026			
	72930 142.85	13.107.4.50	192.168.0.5	TCP	1514 80 → 56690 [ACK] Seq=17523609 Ack=111327 Win=1026			
	72931 142.85	192.168.0.5	13.107.4.50	ТСР	54 56690 → 80 [ACK] Seq=111327 Ack=17525069 Win=1006			
	72932 142.85	13.107.4.50	192.168.0.5	TCP	1514 80 → 56690 [ACK] Seq=17525069 Ack=111327 Win=1026			
	72933 142.85	13.107.4.50	192.168.0.5	TCP	1514 80 → 56690 [ACK] Seg=17526529 Ack=111327 Win=1026			
	72934 142.85	192,168,0,5	13,107,4,50	TCP	54 56690 → 80 [ACK] Seg=111327 Ack=17527989 Win=1006			
	72935 142 85	13 107 4 50	192 168 0 5	TCP	1514 80 - 56690 [ACK] Seg=17527989 Ack=111327 Win=1026			
	72026 142.85	12 107 4 50	102 169 0 5		922 HTTP/1 1 206 Dantial Contont			
	- 72550 142.65	15.107.4.50	192.100.0.5	ппр	025 HITF/1.1 200 Failtar COntent			

Figure 5. ATM Start-up Authentication Packet Capture

With this initial capture, the authentication to the server can be found, along with the resolved IP address of the server, TCP handshake, and TLSv1.2 Client Hello and other information as seen in Figure 6.

4	ATM Startup Filtered.pcapng								
Eile	File Edit View Go Capture Analyze Statistics Telephony. Wireless Iools Help								
	📶 🔳 🔬 🐵 🖡 🖆 🕅 🙆 🔍 🗰 🗯 著 👤 🚃 🔍 🔍 🔍 🏛								
	Apply a display filter <ctrl-></ctrl->								
No.	Time	Source	Destination	Protocol	Length Info				
	1 0.0000	192.168.0.1	192.168.0.7	DHCP	342 DHCP Offer - Transaction ID 0x61dd1155				
	2 0.6384	192.168.0.1	192.168.0.7	DHCP	342 DHCP Offer - Transaction ID 0x61dd1155				
	3 0.6490	192.168.0.1	192.168.0.7	DHCP	342 DHCP ACK - Transaction ID 0x61dd1155				
	4 39,458	192.168.0.1	192.168.0.7	DHCP	342 DHCP Offer - Transaction ID 0x61dd1156				
	5 39,490	192.168.0.1	192.168.0.7	DHCP	342 DHCP ACK - Transaction ID 0x61dd1156				
	6 116.85	192.168.0.7	192.168.0.1	DNS	81 Standard query 0x0051 A EFTDEBITATM.FNFIS.COM				
	7 116.90	192.168.0.1	192.168.0.7	DNS	97 Standard query response 0x0051 A EFTDEBITATM.FNFIS.COM A 206.71.17.21				
	8 116.91	192.168.0.7	206.71.17.21	ТСР	62 49164 → 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK_PERM=1				
	9 116.94	206.71.17.21	192.168.0.7	ТСР	60 443 → 49164 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1436				
	10 116.94	192.168.0.7	206.71.17.21	TCP	60 49164 → 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0				
	11 117.47	192.168.0.7	206.71.17.21	TLSv1.2	361 Client Hello				
	12 117.50	206.71.17.21	192.168.0.7	TLSv1.2	1514 [TCP Previous segment not captured] , Ignored Unknown Record				
	13 117.50	206.71.17.21	192.168.0.7	TLSv1.2	392 Ignored Unknown Record				
	14 117.50	192.168.0.7	206.71.17.21	тср	60 [TCP Dup ACK 10#1] 49164 → 443 [ACK] Seq=308 Ack=1 Win=33028 Len=0				
	15 117.51	206.71.17.21	192.168.0.7	TLSv1.2	396 Server Key Exchange, Server Hello Done				
	16 117.51	192.168.0.7	206.71.17.21	тср	60 [TCP Dup ACK 10#2] 49164 → 443 [ACK] Seq=308 Ack=1 Win=33028 Len=0				
	17 117.55	206.71.17.21	192.168.0.7	тср	1514 [TCP Retransmission] 443 → 49164 [PSH, ACK] Seq=1 Ack=308 Win=35381 Le				
	18 117.55	192.168.0.7	206.71.17.21	ТСР	60 49164 → 443 [ACK] Seq=308 Ack=3601 Win=32348 Len=0				
	19 118.29	192.168.0.7	206.71.17.21	TLSv1.2	180 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message				
	20 118.32	206.71.17.21	192.168.0.7	TCP	60 443 → 49164 [ACK] Seq=3601 Ack=434 Win=35255 Len=0				
<	24.440.22	200 74 47 24	400 400 0 7	TLOUG O	where the state of				
•				1 (200					
>	> Frame 1: 342 bytes on wire (2736 bits), 342 bytes captured (2736 bits) on interface 0								

Figure 6. Initial ATM Authentication Packet Capture

This 22-packet data exchange includes updating the processor with the current terminal number, amount of cash in the vault and any other parameters that may have changed. These could include the terminal ID, software version, and firmware versions for any of the peripherals such as the printer, cash dispensing unit (CDU), card reader, or PCI keypad. In the case of the terminal ID or the PCI Keypad, if they had been changed and not re-authenticated by keying in 3DES keys and a secondary authorization to the server via a POTS connection, the machine would not be authenticated and would not enter a transaction ready state and would go out of order. This machine was properly programmed and updated and was allowed to enter a ready state for customers to initiate a transaction.

After the ATM had been verified to be authenticated to the processing server, the test transactions were run. From the initial capture showing the authentication to the server located at IP address of 206.71.17.21, and the ATM had an address of

192.168.0.7. These addresses were then used to filter the packets in each capture to an output that only included packets from each transaction and those addresses. Once the ATM had received DHCP services and dynamically obtained a private IP address, the start-up data was collected showing that IP address. Then the ATM was statically configured with that same private IP address to keep the data collected consistent and efficient. It should be noted that one of the changes from the captures done in 2012 is that the ATM is programmed to connect to a URL requiring a DNS resolution of that URL to an IP address. In the past, ATM's were programmed with the public IP address of the FIS server which removed the step of the DNS resolution. This was one of several new security protocols implemented at the time that EMV cards were required in ATMs. One of those filtered data captures is shown in Figure 7. The filtered captures for each of the six different transactions can be found in the index.

🚄 Balan	E Balance Check Incorrect PIN Filtered.pcapng							
Eile Ed	lit ⊻iew <u>G</u> o <u>C</u> ap	oture Analyze Statistics	Telephony Wireless Iools Help					
📕 🖩 🖉 🐵 📙 🖹 🗶 🕼 🐧 🗰 🌞 🧱 🖗 🐨 🔜 🔄 🔍 🔍 🖽								
Apply Apply	a display filter <ctrl·></ctrl·>							
No.	Time	Source	Destination	Protocol	Length Info			
F	1 0.0000	192.168.0.7	206.71.17.21	TCP	62 49165 → 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK_PERM=1			
	2 0.1039	206.71.17.21	192.168.0.7	TCP	60 443 → 49165 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1436			
	3 0.1118	192.168.0.7	206.71.17.21	тср	60 49165 → 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0			
	4 0.3812	192.168.0.7	206.71.17.21	TLSv1.2	361 Client Hello			
	5 0.4125	206.71.17.21	192.168.0.7	TCP	1514 443 → 49165 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460 [TCF			
	6 0.4137	206.71.17.21	192.168.0.7	TCP	1514 443 → 49165 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [
	7 0.4140	206.71.17.21	192.168.0.7	TLSv1.2	392 Server Hello, Certificate			
	8 0.4145	192.168.0.7	206.71.17.21	TCP	60 49165 → 443 [ACK] Seq=308 Ack=2921 Win=33028 Len=0			
	9 0.4266	206.71.17.21	192.168.0.7	TLSv1.2	396 Server Key Exchange, Server Hello Done			
	10 0.4269	192.168.0.7	206.71.17.21	TCP	60 49165 → 443 [ACK] Seq=308 Ack=3601 Win=32348 Len=0			
	11 0.9059	192.168.0.7	206.71.17.21	TLSv1.2	180 Client Key Exchange, Change Cipher Spec, Encrypted Handshake			
	12 0.9454	206.71.17.21	192.168.0.7	TCP	60 443 → 49165 [ACK] Seq=3601 Ack=434 Win=35255 Len=0			
	13 0.9454	206.71.17.21	192.168.0.7	TLSv1.2	105 Change Cipher Spec, Encrypted Handshake Message			
	14 0.9800	192.168.0.7	206.71.17.21	TLSv1.2	731 Application Data			
	15 1.0034	206.71.17.21	192.168.0.7	TCP	60 443 → 49165 [ACK] Seq=3652 Ack=1111 Win=39848 Len=0			
	16 1.5069	206.71.17.21	192.168.0.7	TLSv1.2	221 Application Data			
	17 1.6395	192.168.0.7	206.71.17.21	TCP	60 49165 → 443 [ACK] Seq=1111 Ack=3819 Win=32130 Len=0			
	18 1.7103	192.168.0.7	206.71.17.21	TLSv1.2	85 Encrypted Alert			
	19 1.7889	206.71.17.21	192.168.0.7	TCP	60 443 → 49165 [ACK] Seq=3819 Ack=1142 Win=39817 Len=0			
	20 1.8013	192.168.0.7	206.71.17.21	TCP	60 49165 → 443 [FIN, ACK] Seq=1142 Ack=3819 Win=32130 Len=0			
	21 1.8258	206.71.17.21	192.168.0.7	TCP	60 443 → 49165 [FIN, ACK] Seq=3819 Ack=1143 Win=39817 Len=0			
	22 1.8260	192.168.0.7	206.71.17.21	TCP	60 49165 → 443 [ACK] Seq=1143 Ack=3820 Win=32130 Len=0			
> Ena	ame 1: 62 by	tes on wire (49	6 bits), 62 bytes capture	d (496 bi	ts) on interface 0			
Thome I, by bytes on white (450 bits), by bytes capterial (550 bits) on interface 0								
Toternet Protocol Varsion A Sec. 102.168.0.7 (00.007/0.301.00.7) 36: McCgeal_01.05.98 (76.02.94.01.05.98)								
Transmission Control Protocol Sec Port: 443 Sec. 0 Jon: 0								
	Transmission control Protocol, Sic Port. 49105, USC Port. 445, Seq: 0, Len: 0							
000	0 78 d2	94 6f d3 9a	00 a0 7c 31 5d 37	08 00 4	15 00			
0.00								

Figure 7. Filtered Packet Capture

For comparison to the previous work which is the basis for the current research questions, Table 1 shows the packet numbers and size for each transaction in the 2012 study. Every transaction type was completely predictable in the number of packets and only had two outliers in packet size out of 85 data points. This confirmed the possibility of creating a network forensics benchmark for ATM Ethernet traffic under the conditions tested and the authenticating server's requirements at that time.

Table 1

Packet Number	Test 1	Test 2	Test 3	Test 4	Test 5
1	60	62	62	60	60
2	60	60	60	60	60
3	60	60	60	60	60
4	269	269	269	269	269
5	793	793	793	793	793
6	60	60	60	60	60
7	244	244	244	244	244
8	60	60	60	60	60
9	105	105	105	105	105
10	320	320	320	320	320
11	60	60	60	60	60
12	187	187	187	187	187
13	60	60	60	60	60
83	83	83	83	83	83
15	60	60	60	60	60
16	60	60	60	60	60
17	60	60	60	60	60

Packets Captured for Five Test Transactions

However, Table 2 created from the current research shows a different result. The number of packets per transaction varies from 21 to 24, and the orderly size of packets found in the original research is not found here either.

Table 2

Packet #	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
1	62	62	62	62	62	62
2	60	60	60	60	60	60
3	60	60	60	60	60	60
4	361	361	361	361	361	361
5	1514	1514	1514	1514	1514	1514
6	1514	1514	1514	1514	1514	1514
7	392	392	392	392	392	392
8	60	396	60	60	60	60
9	105	60	396	396	396	396
10	396	180	60	60	60	60
11	60	60	180	180	180	180
12	187	105	60	60	60	60
13	60	731	105	105	105	105
14	180	60	731	60	60	731
15	60	221	60	731	731	60
16	105	60	221	60	60	221
17	731	85	60	221	221	85
18	60	60	85	85	60	60
19	221	60	60	60	85	60
20	60	60	60	60	60	60
21	60	60	60	60	60	60
22	60		60	60	60	
23	60				60	
24	60					

Packets Captured for Five Test Transactions (2019)

By taking a closer look at the composition of the packets, some assumptions can be made to cull the data a bit further into something more useful for this research. Table 2 shows that the packets with lengths of 60 and 62 can be considered to be overhead packets. They are acknowledgments to the server for specific frames outlined in red in Figure 8 and can be considered extraneous for this research. The variance in the number of these packets is not a product of the different types of transactions process (i.e., withdrawal, vs. balance check) but varying network infrastructure conditions between the ATM and the processing server that may require the re-transmittal of a packet or a duplicate acknowledgment. This is seen with more clarity when looking at the ATM start-up packet capture which can be seen in Table 6 and is discussed further in this paper. These are indicated by the light blue background in the cells of Table 2.

```
Successful Withdrawl 20 Filtered.pcapng
<u>File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help</u>
🛋 🔳 🖉 🐵 📜 🛅 🕺 🗳 I 🤇 🗢 🌧 警 🖌 💆 📃 🖳 Q. Q. Q. 🎞
Ar
        Tim
                                                          Protoco
                                                                       Infe
       1 0.0000... 192.168.0.7
                                206.71.17.21
                                                          TCP
                                                                     62 49169 → 443 [SYN] Seq=0 Win=32768 Len=0 MSS=146
      2 0.0316... 206.71.17.21 192.168.0.7
                                                          TCP
                                                                     60 443 → 49169 [SYN, ACK] Seq=0 Ack=1 Win=8190 Ler
      3 0.0317... 192.168.0.7 206.71.17.21
                                                                     60 49169 → 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0
                                                          ТСР
       4 0,3068... 192,168,0,7
                                206.71.17.21
                                                          TLSv1.2
                                                                   361 Client Hello
      5 0.3374... 206.71.17.21
                               192,168,0,7
                                                          TCP
                                                                   1514 443 → 49169 [PSH, ACK] Seq=1 Ack=308 Win=35381
      6 0.3386... 206.71.17.21 192.168.0.7
                                                          TCP
                                                                   1514 443 → 49169 [PSH, ACK] Seq=1461 Ack=308 Win=353
       7 0.3388... 206.71.17.21
                                192.168.0.7
                                                          TLSv1.2 392 Server Hello, Certificate
      8 0.3393... 192.168.0.7
                                                          ТСР
                                                                    60 49169 → 443 [ACK] Seq=308 Ack=2921 Win=33028 Le
                                206.71.17.21
      9 0.3475... 206.71.17.21 192.168.0.7
                                                          TLSv1.2 396 Server Key Exchange, Server Hello Done
     10 0.3477... 192.168.0.7
                                206.71.17.21
                                                          TCP
                                                                    60 49169 → 443 [ACK] Seq=308 Ack=3601 Win=32348 Le
     11 0.6246... 192.168.0.7
                                206.71.17.21
                                                          TLSv1.2 180 Client Key Exchange, Change Cipher Spec, Encryp
    [TCP Segment Len: 0]
    Sequence number: 1 (relative sequence number)
    [Next sequence number: 1 (relative sequence number)]
    Acknowledgment number: 1
                                 (relative ack number)
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x010 (ACK)
    Window size value: 33028
    [Calculated window size: 33028]
     [Window size scaling factor: -2 (no window scaling used)]
    Checksum: 0x8ca3 [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  ✓ [SEQ/ACK analysis]
       [This is an ACK to the segment in frame: 2]
       [The RTT to ACK the segment was: 0.000138000 seconds]
       [iRTT: 0.031770000 seconds]
  > [Timestamps]
```

Figure 8. Overhead Packet

When the data is culled to remove the overhead packets, the new table once

again shows the predictability of ATM transaction traffic on an Ethernet network. What is

left are the important packets. Table 3 shows the six test transactions and the important packets that make up each transaction.

Table 3

Packet #	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
	00	00	00	00	00	00
1	62	62	62	62	62	62
2	60	60	60	60	60	60
3	60	60	60	60	60	60
4	361	361	361	361	361	361
5	1514	1514	1514	1514	1514	1514
6	1514	1514	1514	1514	1514	1514
7	392	392	392	392	392	392
8	396	396	396	396	396	396
9	180	180	180	180	180	180
10	105	105	105	105	105	105
11	731	731	731	731	731	731
12	221	221	221	221	221	221
13	85	85	85	85	85	85

Table 4 shows the protocol used, length, and other information about each of the important packets. The packet number and time stamp were removed to allow enough space to display the relevant information. The raw data and truncated tables are available in the index.

Table 4

Typical Transaction Packet Data and Information

Source	Destination	Protocol	Length	Info
192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
206.71.17.21	192.168.0.7	ТСР	1514	443 > 49165 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460 [TCP segment of a reassembled PDU]
206.71.17.21	192.168.0.7	ТСР	1514	443 > 49165 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TCP segment of a reassembled PDU]
206.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certificate
206.71.17.21	192.168.0.7	TLSv1.2	396	Server Key Exchange, Server Hello Done
192.168.0.7	206.71.17.21	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
206.71.17.21	192.168.0.7	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message
192.168.0.7	206.71.17.21	TLSv1.2	731	Application Data
206.71.17.21	192.168.0.7	TLSv1.2	221	Application Data
192.168.0.7	206.71.17.21	TLSv1.2	85	Encrypted Alert

Data Analysis

To answer each of the research questions, each data capture was filtered to only include the packets relevant to the transaction between the ATM and the processing server. For each of the six transactions, the overhead packets were removed leaving only those packets that processed the transaction. Then the packets for each transaction were compared with regards to their number, size, and order processed. The results can be found in each of the Appendixes. For each of the six transaction types, the appendix includes the Wireshark capture as well as an exported CSV file displayed in Microsoft Excel. Each of the CSV files were then manipulated to remove the overhead packets. Then each of the transaction's packets were compared and this is found specifically in Appendix H.

Chapter V: Results, Conclusion, and Recommendations

Introduction

This research is intended to discover the packet characteristics of Ethernet traffic from a common retail ATM. The results of the research show both some expected results and interesting changes from previous work.

Results

The answer to each of the study questions begins with looking at the general transaction packets. Each transaction, when overhead packets are removed from the analysis, was constructed of 18 packets. The initial SYN-ACK packets, which could be considered overhead, are included in the table below since they are required to set up the communication. Table 5 shows the packets considered for the analysis.

Table 5

	Final	Packet	for A	nal	vsis
--	-------	--------	-------	-----	------

Source	Destination	Protocol	Length	Info
192.168.0.7	206.71.17.21	ТСР	62	49165 > 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK_PERM=1
206.71.17.21	192.168.0.7	ТСР	60	443 > 49165 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1436
192.168.0.7	206.71.17.21	ТСР	60	49165 > 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0
192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
206.71.17.21	192.168.0.7	ТСР	1514	443 > 49165 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460 [TCP segment of a reassembled PDU]
206.71.17.21	192.168.0.7	ТСР	1514	443 > 49165 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TCP segment of a reassembled PDU]
206.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certificate
206.71.17.21	192.168.0.7	TLSv1.2	396	Server Key Exchange, Server Hello Done
192.168.0.7	206.71.17.21	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
206.71.17.21	192.168.0.7	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message
192.168.0.7	206.71.17.21	TLSv1.2	731	Application Data
206.71.17.21	192.168.0.7	TLSv1.2	221	Application Data
192.168.0.7	206.71.17.21	TLSv1.2	85	Encrypted Alert

For each of the six transaction types in Study Question 1, the packet construction was consistent without deviation of non-overhead packets. Because of the predictability of the packet lengths, no matter the transaction type, it creates the possibility of created a forensics benchmark for ATM Ethernet traffic. The combination of 18 packets with packet lengths of 62, 60, 60, 361, 1514, 1514, 392, 396, 180, 105, 731, 231, and 85

should produce a profile with a high probability of identifying this specific traffic. The results for each research question are as follows:

- 1. How many packets make up each of the 6 transactions types?
 - a. There are 13 packets of consistent length that made up each transaction.
 Refer to Appendixes A through F.
- 2. Is each packet of predictable length?
 - a. Yes, when the overhead packets were removed, the packets were consistent in length. Refer to Appendix H.
- 3. Is packet length influenced by transaction type?
 - a. No, there is not any variance in packet length noted from one transaction type to another. Refer to Appendix H
- 4. What protocols were used throughout each transaction?
 - a. The TCP protocol was used for each transaction with TLSv1.2 used for encryption and data integrity. Refer to Appendix A through H for the data captures that include the protocols use for each packet and each transaction.
- 5. What known vulnerabilities exist regarding the packet construction and protocols used?
 - a. Packet construction predictable packet length may be vulnerable to a Teardrop attack, especially with the fragmented packets seen in the packet captures, most notably in the ATM start up capture as seen in Table 6. As noted in an article in Security and Communications Networks,

low-rate DDoS attacks are configured often to use know packet lengths used by a service or service to mask the attack. It was also suggested that this could make the attack more difficult to discover in logs because the traffic has the many of the characteristics of expected packets. "Low-rate Distributed Denial-of-Service (low-rate DDoS) attacks are a new challenge to cyberspace, as the attackers send a large amount of attack packets similar to normal traffic, to throttle legitimate flows. (Zhou, Liao, Yuan, & Zhang, 2017, p. 1).

- b. TLSv1.2 has at least six know vulnerabilities: (Prodromou, 2019).
 - i. The Padding Oracle on Downgraded Legacy Encryption (POODLE) attack was published in October 2014 and takes advantage that some servers/clients still support SSL 3.0 for compatibility with legacy systems. There are several named vulnerabilities or offshoots of the POODLE name. Zombie POODLE and GOLDENPOODLE were presented at Black Hat Asia last month.

Similar to ROBOT, DROWN and many other vulnerabilities affecting HTTPS, these issues stem from continued use of cryptographic modes which should have been long ago deprecated and yet are inexplicably still supported in TLSv1.2. In this case, the troublesome feature is that TLSv1.2 supports CBC mode ciphersuites (Young, 2019).

- ii. The Browser Exploit Against SSL/TLS (BEAST) was disclosed in September 2011 and applies to SSL 3.0 and TLS 1.0 so it affects browsers that support TLS1.0 or earlier protocols. The same mitigation efforts for the POODLE vulnerability apply to BEAST.
- iii. The Compression Ration Info-leak Made Easy (CRIME) vulnerability affects TLS compression. The compression method is included in the Client Hello message (see Table 6) and it is optional. It is possible to establish a connection without compression. This appears to be a browser related vulnerability and may not apply to this research. However, it will be mentioned in future work.
- iv. The Browser Reconnaissance and Exfiltration via Adaptive Compression of Hypertext (BREACH) vulnerability is very similar to CRIME but BREACH targets HTTP compression, not TLS compression. This attack is possible even if TLS compression is turned off. As with CRIME, BREACH does not appear to be a vulnerability in this scenario.
- v. Heartbleed was a critical vulnerability that was found in the heartbeat extension of the popular OpenSSL library. This extension was used by the previous version of the retail ATM tested and is no longer part of the threat landscape.

For the ATM startup packet, there are some similarities in the number and size of packets. However, the data collected possibly shows some effects of the congested network segment to which the ATM was connected. As seen in Table 6, there are multiple packets that were re-transmitted, and segments not captured shown in cells with type in red. However, it seems that the process of the initial startup communications with the processing server is at least in part, duplicated with each processed transaction.

Table 6

ATM Start-up Capture

No.	Source	Destination	Protocol	Length	Info
1	192.168.0.1	192.168.0.7	DHCP	342	DHCP Offer - Transaction ID 0x61dd1155
2	192.168.0.1	192.168.0.7	DHCP	342	DHCP Offer - Transaction ID 0x61dd1155
3	192.168.0.1	192.168.0.7	DHCP	342	DHCP ACK - Transaction ID 0x61dd1155
4	192.168.0.1	192.168.0.7	DHCP	342	DHCP Offer - Transaction ID 0x61dd1156
5	192.168.0.1	192.168.0.7	DHCP	342	DHCP ACK - Transaction ID 0x61dd1156
6	192.168.0.7	192.168.0.1	DNS	81	Standard query 0x0051 A EFTDEBITATM.FNFIS.COM
7	192.168.0.1	192.168.0.7	DNS	97	Standard query response 0x0051 A EFTDEBITATM.FNFIS.COM A 206.71.17.21
8	192.168.0.7	206.71.17.21	ТСР	62	49164 > 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK_PERM=1
9	206.71.17.21	192.168.0.7	ТСР	60	443 > 49164 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1436
10	192.168.0.7	206.71.17.21	ТСР	60	49164 > 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0
11	192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
12	206.71.17.21	192.168.0.7	TLSv1.2	1514	[TCP Previous segment not captured] , Ignored Unknown Record
13	206.71.17.21	192.168.0.7	TLSv1.2	392	Ignored Unknown Record
14	192.168.0.7	206.71.17.21	ТСР	60	[TCP Dup ACK 10#1] 49164 > 443 [ACK] Seq=308 Ack=1 Win=33028 Len=0
15	206.71.17.21	192.168.0.7	TLSv1.2	396	Server Key Exchange, Server Hello Done
16	192.168.0.7	206.71.17.21	ТСР	60	[TCP Dup ACK 10#2] 49164 > 443 [ACK] Seq=308 Ack=1 Win=33028 Len=0
17	206.71.17.21	192.168.0.7	ТСР	1514	[TCP Retransmission] 443 > 49164 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460

Table 6 Continued

No.	Source	Destination	Protocol	Length	Info
18	192.168.0.7	206.71.17.21	ТСР	60	49164 > 443 [ACK] Seq=308 Ack=3601 Win=32348 Len=0
19	192.168.0.7	206.71.17.21	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
20	206.71.17.21	192.168.0.7	ТСР	60	443 > 49164 [ACK] Seq=3601 Ack=434 Win=35255 Len=0
21	206.71.17.21	192.168.0.7	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message
22	192.168.0.7	206.71.17.21	TLSv1.2	221	Application Data
23	206.71.17.21	192.168.0.7	ТСР	60	443 > 49164 [ACK] Seq=3652 Ack=601 Win=40358 Len=0
24	206.71.17.21	192.168.0.7	TLSv1.2	150	Application Data
25	192.168.0.7	206.71.17.21	ТСР	60	49164 > 443 [ACK] Seq=601 Ack=3748 Win=32201 Len=0
26	192.168.0.7	206.71.17.21	TLSv1.2	85	Encrypted Alert
27	192.168.0.7	206.71.17.21	ТСР	60	49164 > 443 [FIN, ACK] Seq=632 Ack=3748 Win=32201 Len=0
28	206.71.17.21	192.168.0.7	ТСР	60	443 > 49164 [ACK] Seq=3748 Ack=632 Win=40327 Len=0
29	206.71.17.21	192.168.0.7	ТСР	60	443 > 49164 [FIN, ACK] Seq=3748 Ack=633 Win=40327 Len=0
30	192.168.0.7	206.71.17.21	ТСР	60	49164 > 443 [ACK] Seq=633 Ack=3749 Win=32201 Len=0

Conclusion

The conclusions of this research are surprisingly similar to those of previous work in that the transactions were predictable in packet numbers, size and sequence. In the view of the author, this presents two possibilities. The predictability can be used in forensics efforts to identify ATM traffic should packet inspection be needed to answer questions in an investigation of a compromised ATM. Secondly, it could present an interesting and not commonly known attack surface to be exploited.

Future Work

The results of this research have identified several areas of future work some of which would require gaining permission from the credit card issuing entities who own much of the proprietary infrastructure that retail ATMs connect too.

Beginning with the ATM startup capture, additional packet captures should be run to get a more accurate picture of what packets consistently make up the initial communication with the processing server. It should also be analyzed to determine what similarities if any exists between the startup communications packets and those that make up a customer-initiated transaction.

When choosing the transactions to test, two similar transactions were chosen intentionally with a minor change. For instance, two successful withdrawals but with different dollar amounts were processed. This was done with the idea of future work in mind if permission can be obtained for packet payload inspection and other efforts to identify meaningful data in each packet. In previous work it was noted that packets can contain human-readable data related to the processor.

It would also be useful to conduct cable tapping testing as Ethernet cables often exit the vault portion of the ATM and are easily accessible in a retail setting. Also, it is very common to find that the Ethernet cable servicing the ATM is cut, re-terminated to allow for a switch or hub to be placed near the machine to allow for the connection of other machines such as digital jukeboxes or other internet connected devices, now so ubiquitous in the retail environment. Lastly, the POODLE and compressions related vulnerabilities should be considered if permission for in-depth packet inspection were obtained. If compression is not used, there seems to be the potential for nefarious actors to, at a minimum, gain actionable intel or at the maximum, compromise a transaction or even be able to redirect traffic to a compromised machine.

References

- ADA National Network. (n.d.). *Questions, help, and training on the ADA*. Retrieved from www.adaresources.org/
- ATM of America. (n.d.). *Hyosung 1800 manual*. Retrieved from www.atmofamerica.com/Manuals/Hyosung/NH1800.pdf
- Adepju, A. S., & Mohammed, E. A. (2010). Challenges of automated teller machine (atm) usage and fraud occurrences in Nigeria: A case study of selected banks in Minna Metropolis. *Journal of Internet Banking and Commerce, 15*(2), 1-10.
 Retrieved from www.arraydev.com/commerce/jibc/.
- Adeoye, O. (2012). Evaluating the performance of two-factor authentication solution in the banking sector. *International Journal of Computer Sciences Issues*, 9(4), 457-462.
- Alblawi, U., & Jinoh, K. (2014). A hybrid classifier with a binning method for network application identification. *Journal of Integrated Design and Process Science,* 18(3), 3-22. Retrieved from https://www.researchgate.net/publication/279709222_A_Hybrid_Classifier_with_ a_Binning_Method_for_Network_Application_Identification
- Alshammari, R., & Zincir-Heywood, A. N. (2010). "Can encrypted traffic be identified without port numbers, IP addresses and payload inspection? *Computer Networks*, *55*(6), 1326-1350. doi:10.1016/j.comnet.2010.12.002
- Automated teller machine. (n.d.). In *Merriam Webster*. Retrieved from https://www.merriam-webster.com/dictionary/automated%20teller%20machine

- Banerjee, U., Vashishtha, A., & Saxena, M. (2010). Evaluation of the capabilities of wireshark as a tool for intrusion detection. *International Journal of Computer Applications*, 6(7), 1-5. doi:10.5120/1092-1427.
- Bluetooth. (n.d.) In *English Oxford Living Dictionaries*. Retrieved from https://en.oxforddictionaries.com/definition/bluetooth.
- Brunner, A., Decressin, J., Hardy, D., & Kudela, B. (2004). Germany's three-pillar banking system: Cross-country perspectives in Europe. Washington, DC:
 International Monetary Fund.
- Chafalon, S. (2012). *Bank fraud and ATM security*. Retrieved from http://resources.infosecinstitute.com/bank-fraud-atm-security/
- Cisco Networking Academy. (2017). *Routing and switching essentials*. Hoboken, NJ: Cisco Press, 2017.
- Debnath, R., Agrawal, P., & Vaishnav, G. (2014). Des, aes and triple des: Symmetric key cryptography algorithm. *International Journal of Science, Engineering and Technology Research*, 3(3), 652-654. Retrieved from ijsetr.org/wpcontent/uploads/2014/03/IJSETR-VOL-3-ISSUE-3-652-654.pdf
- Firewall. (n.d.) In *English Oxford Living Dictionaries*. Retrieved from https://en.oxforddictionaries.com/definition/firewall.
- Hossain, M. S., Russel, A. H., & Robidas, L. C. (2015). Analysis of factors affecting the customer's satisfaction with reference to ATM services in Dhaka City. *IOSR Journal of Business and Management*, 17(11) 1, 68-75.

Khan, F. (2015). *How does an ATM machine work?* Retrieved from www.quora.com/How-does-an-ATM-machine-work-1

Modern Ghana. (2013). *Hackers steal \$45 million in ATM card scam, federal prosecutors say.* Retrieved from www.modernghana.com/news/463043/1/hack ers-steal-45-million-in-atm-card-scam-federal.html

Plain old telephone service. (n.d.). In *Wikipedia*. Retrieved from https://en.wikipedia.org/wiki/Plain_old_telephone_service.

- Prodromou, A. (2019). *TLS security 6: Examples of TLS vulnerabilities and attacks*. Retrieved from https://www.acunetix.com/blog/articles/tls-vulnerabilities-attacks-finalpart/143-153
- Pymnts.com. (2017). US ATMs finally see growth. Retrieved from https://www.pymnts.com/cash/2017/atmia-says-us-atms-finally-see-growth/
- Rupam, A., Verma, A., & Singh, A. (2013). An approach to detect packets using packet sniffing. *International Journal of Computer Science and Engineering Survey*, *4*(3), 21-33. doi:10.5121/ijcses.2013.4302.
- Sidel, R. (2015). *Theft of debit-card data from ATMs soars*. Retrieved from www.wsj.com/articles/theft-of-debit-card-data-from-atms-soars-1432078912
- Souvignet, T., Hatin, J., Maqua, F., Tesniere, D., Léger, P., & Hormière, R. (2014). Payment card forensic analysis: From concepts to desktop and mobile analysis tools. *Digital Investigation, 11*(3), 143-153.

TechTarget Network. (n.d.a). *What is ethernet?* Retrieved from https://searchnetworking.techtarget.com/definition/Ethernet.

- TechTarget Network. (n.d.b). *What is local area network (LAN)*? Retrieved from searchnetworking.techtarget.com/definition/local-area-network-LAN
- Twum, F., Nti, I. K., & Asante, M. (2016). Improving security levels in automatic teller machines (ATM) using multifactor authentication. *International Journal of Science* and Engineering Applications, 5(3), 126-134. doi:10.7753/ijsea0503.1003.
- Wireshark. (n.d.). In *Techopedia*. Retrieved from www.techopedia.com/definition/25325/wireshark
- Young, C. (2019). *Introducing Zombie POODLE and GOLDENDOODLE*. Retrieved from https://www.tripwire.com/state-of-security/vulnerability-management/zombiepoodle-goldendoodle/
- Zhou, L., Liao, M., Yuan, C., & Zhang, H. (2017). Low-rate DDoS attack detection using expectation of packet size. Security and Communication Networks, 2017(1), 1-14. doi:10.1155/2017/3691629

Appendices

Appendix A Successful Withdrawal \$20

\bigcirc	0	\sim	\sim	\sim	\sim	Г																							No.			Fie	
🕈 Succesfu	000 7	Transmis	Internet	Ethernet	Frame 1:	231.	22 1.	21 1.	201.	191.	181.	17 1.	160.	150.	140.	130.	120.	110.	100.	.06	80.	70.	60.	50.	40.	30.	20.	10.	Te	Apply a display fib		<u>E</u> dit <u>V</u> iew	uccessful With
Wehdrawl	8 d2	sion (Prote	н,	62 by	6596	6594	6095	5861	5828	4195	3154	8421	8141	8134	6683	6485	6246	3477	3475	3393	3388	3386	3374	3068	0317	0316	0000	•	r <ctrl- ;<="" td=""><td></td><td>ی ۵</td><td>drawl 20 Fi</td></ctrl->		ی ۵	drawl 20 Fi
20 Filtered.pc	94 6	Contro	ocol V	irc: T	/tes o	192.1	206.7	206.7	192.1	192.1	192.1	206.7	206.7	192.1	192.1	206.7	206.7	192.1	192.1	206.7	192.1	206.7	206.7	206.7	192.1	192.1	206.7	192.1	Source			pture <u>A</u> r	tered.pcap
bude	f d3	l Prot	ersion	onyang	n wire	68.0.	1.17.	1.17.	168.0.	168.0.	168.0.	1.17.	1.17.	68.0.	68.0.	1.17.	1.17.	68.0.	68.0.	1.17.	68.0.	1.17.	1.17.	1.17.	68.0.	68.0.	1.17.	168.0.			<u>م</u>	alyze <u>S</u> ta	ŋ
	9a 0	ocol,	4, Sr	N_31:5	(496	2	21 1	21 1	2	2	2	21 1	21 1	2	2	21 1	21 1	2	2	21 1	7	21 1	21 1	21 1	2	2	21 1				*	tistics Te	
	0 a0	Shc Po	'C: 193	d:37	bits)	06.71	192.16	92.16	06.71	06.71	06.71	92.16	92.16	06.71	06.71	92.16	92.16	06.71	06.71	92.16	06.71	92.16	92.16	92.16	06.71	06.71	192.16	06.71	Vestination) 	elephony	
	7c	ort: 4	2.168.	(00:a0	, 62 b	.17.21	8.0.7	8.0.7	.17.21	.17.21	.17.21	8.0.7	8.0.7	.17.21	.17.21	8.0.7	8.0.7	.17.21	.17.21	8.0.7	.17.21	8.0.7	8.0.7	8.0.7	.17.21	.17.21	8.0.7	.17.21				<u>W</u> ireless	
	ы С	9169,	0.7, [:7c:31	ytes o																										₽ □	Tools H	
	d 37	Dst Po)st: 20	:5d:37	apture																										₽	놓	
	80	ort: 4	36.71.	7), Dst	ed (49	TCP	TCP	ТСР	TCP	TLSV	TCP	TLSV	TCP	TLSV	TCP	TLSv	TCP	TLSV	TCP	TLSV	TCP	TLSV	TCP	TCP	TLSV	TCP	TCP	ТСР	Protocol				
	0 45	13, Se	17.21	t: Net	5 bits					1.2		1.2		1.2		1.2		1.2		1.2		1.2	1		1.2				5				
	00	q: 0,		gear_6) on i	60 49	60 44	60 44	60 49	85 En	60 49	221 Ap	60 44	731 Ap	60 49	105 Ch	60 44	180 Cl	60 49	396 Se	60 49	392 Se	514 44	514 44	361 Cl	60 49	60 44	62 49	ingth Info				
		Len: (f:d3:9	nterfa	169 →	3 + 49	$3 \rightarrow 49$	169 →	crypte	169 →	plicat	3 → 49	plicat	169 +	ange (3 → 49	ient k	169 →	rver k	169 →	rver H	3 → 49	3 → 49	ient H	169 →	3 + 49	169 →					
		Ŭ)a (78	ice Ø	443 [/	169 [H	169 [/	443 [F	d Ale	443 [/	ion Da	169 [/	ion Da	443 [/	ipher	169 [/	ey Exc	443 [/	ey Exc	443 [/	ello,	169 [F	169 [H	lello	443 [/	169 [9	443 [9					
				:d2:94		ACK] S	IN, A	ACK] S	IN, A	A	ACK] S	ata	ACK] S	ata	ACK] S	Spec,	ACK] S	change	ACK] S	change	ACK] S	Certi	PSH, Α	SH, A		ACK] S	SYN, A	s [nas					
				:6f:d		eq=114	CK] Se	eq=381	CK] Se		eq=11)		eq=365		eq=434	Encry	eq=366	, Char	eq=308	, Serv	eq=308	ficate	CK] Se	CK] Se		eq=1 /	CK] Se	eq=0 V					
				3:9a)		13 Ack	eq=381	19 Ack	eq=114		1 Ack		2 Ack		I Ack=	/pted)1 Ack	nge Ci	Ack=	rer He	Ack=	10	9q=146	eq=1 A		\ck=1	eq=0 A	lin=32					
						=3820	9 Ack=	=1142	2 Ack=		=3819		=1111		3652 W	Handsh	=434 W	pher s	3601 W	llo Do	2921 W		1 Ack=	ck=308		√in=33	ck=1 W	768 Le					
Packets:						Win=32	1143 1	Win=39	3819 1		Win=32		Win=39		in=322	ake Me	in=352	pec, E	in=323	ne	in=336		308 Wi	Win=		028 Le	in=819	n=0 Ms					
23 · Displaye						130 L	lin=39)817 L	lin=32		130 L		1848 L		97 Le	essage	55 Le	incryp	48 Le)28 Le		.n=353	5381		en=0	0 Len	S=146					
d: 23 (100.0						en=0	817 Le	en=0	130 Le		en=0		en=0		n=0		n=0	ted Ha	n=0		n=0		81 Ler	Len=14			SSW 0=	0 SACK					
6							n=0		n=0									ndshal					=1460	60 [Ti			=1436	PERM					
																		ce Mes					[TCP	CP seg				Ä					
																		sage					segmen	ment o									
																							it of a	if a re									
) reas	assem						- Expr			1
Profile: Defau	()																						sembl	bled						ession +			×

Figure 9. Successful Withdrawal \$20 (Wireshark)

ີ ພີ					5 Withdrawl 2	20 - Excel
File Home Insert Page Layout Formu	las Data	Review V	/iew 🗘 Tell me what y	o want to do		
Calibri + 11 +		₩ ₩ 1	Wrap Text	General		Normal
Paste voly v v ✓ Format Painter B I U v I ⊞ v I 🖄 v	A - ≡ ≡	↓ ↓	Merge & Center 🔹	00° 00° € % ~ \$	Conditional Forr Formatting * Tal	hat as Neutral
Clipboard 🖙 Font	a	Align	iment 5	Number 5		
F26 \checkmark : \times \checkmark fx						
A B C D	m	т		G		
			443 > 49169 [SYN, A	VCK] Seq=0 Ack=1 V	Vin=8190 Len=()
3 2 0.031632 206.71.17.21 192.168.0.7	TCP	60	MSS=1436			
4 3 0.03177 192.168.0.7 206.71.17.2	1 TCP	60	49169 > 443 [ACK] S	eq=1 Ack=1 Win=3	3028 Len=0	
5 4 0.30683 192.168.0.7 206.71.17.2	1 TLSv1.2	361	Client Hello			
6 5 0 337405 206 71 17 21 102 168 0 7	TCp	1514	443 > 49169 [PSH, A	CK] Seq=1 Ack=308	3 Win=35381 Le	n=1460
			443 > 49169 [PSH, A	CK] Seq=1461 Ack=	=308 Win=3538	4
7 6 0.338608 206.71.17.21 192.168.0.7	TCP	1514	Len=1460 [TCP segme	ent of a reassembl	ed PDU]	
8 7 0.338801 206.71.17.21 192.168.0.7	TLSv1.2	392	Server Hello, Certific	ate		
9 8 0.339371 192.168.0.7 206.71.17.2	1 TCP	60	49169 > 443 [ACK] S	eq=308 Ack=2921	Win=33028 Len	0=1
10 9 0.347535 206.71.17.21 192.168.0.7	TLSv1.2	396	Server Key Exchange,	, Server Hello Done		
11 10 0.347736 192.168.0.7 206.71.17.2	1 TCP	60	49169 > 443 [ACK] S	eq=308 Ack=3601	Win=32348 Len	0=1
		2	Client Key Exchange,	Change Cipher Sp	ec, Encrypted H	landshake
12 11 U.02402 192.100.U.7 200.71.17.2 13 10 0.648550 006 71 17 01 100 168 0 7	TCP	50	NU3 < NO160 [NUK] 6	NEV-4-0 1036-021	Win-25255 100	5
14 13 0.668324 206.71.17.21 192.168.0.7	TLSv1.2	105	Change Cipher Spec,	Encrypted Handsh	ake Message	
15 14 0.813417 192.168.0.7 206.71.17.2	1 TCP	60	49169 > 443 [ACK] S	eq=434 Ack=3652	Win=32297 Len	1=0
16 15 0.814157 192.168.0.7 206.71.17.2	1 TLSv1.2	731	Application Data			
17 16 0.842189 206.71.17.21 192.168.0.7	TCP	60	443 > 49169 [ACK] S	eq=3652 Ack=1111	Win=39848 Le	in=0
18 17 1.31549 206.71.17.21 192.168.0.7	TLSv1.2	221	Application Data			
19 18 1.41955 192.168.0.7 206.71.17.2	1 TCP	60	49169 > 443 [ACK] S	eq=1111 Ack=3819) Win=32130 Le	n=0
20 19 1.582867 192.168.0.7 206.71.17.2	1 TLSv1.2	85	Encrypted Alert			
21 20 1.586143 192.168.0.7 206.71.17.2	1 TCP	60	49169 > 443 [FIN, A	CK] Seq=1142 Ack=	3819 Win=3213	30 Len=0
22 21 1.609597 206.71.17.21 192.168.0.7	TCP	60	443 > 49169 [ACK] S	eq=3819 Ack=1142	2 Win=39817 Le	in=0
23 22 1.659424 206.71.17.21 192.168.0.7	TCP	60	443 > 49169 [FIN, A	CK] Seq=3819 Ack=	1143 Win=3981	17 Len=0
24 23 1.659695 192.168.0.7 206.71.17.2	1 TCP	60	49169 > 443 [ACK] S	eq=1143 Ack=3820) Win=32130 Le	in=0
25						
27						
28						
29						
5 Withdrawl 20 (+)						
Ready						

Figure 10. Successful Withdrawal \$20 (CSV)

End Description Water Link Heim Sec of the sec of th	Star Destease Network	0000 00 a0 7c 31 5d 37 78 d2 94 6f d3 9a 08 00 45 00
Bit Bit Way So Carlton Taylong Water Joe Hei Bit Way So Carlton Way Solution Taylong Water Joe Hei Bit Way Solution Way Solutin Way Solutenew Solutan Way Solution Way Solution Way Solution Way	Store Determine Research <	
Effe fan Warden Gauer Mayer Statist Tealowy Warder Jock Hep Image State	Stare Determinant Determinant <th< td=""><td>> Transmission Control Protocol Src Port: 143 Dst Port: 19170 Sec. 0 Ac</td></th<>	> Transmission Control Protocol Src Port: 143 Dst Port: 19170 Sec. 0 Ac
Effe En Wards So Carlon Market Table So Carlon Market Table So Carlon Market Market So Carlon Market Market </td <td>Store Denter Detect TCP Sock Carget Langet Inc Soc. 192.168.0.7 206.71.17.21 192.168.0.7 206.71.17.21 1125.1 60.413.4 49170 443 [SVN] Seq=0 ACK=1 Win=32768 Len=0 MSS=1460 SACK Soc. 192.168.0.7 206.71.17.21 1125.1 Sit [Linet Hell] Sit [Linet Hell] Seq=1 Ack=1 Win=33028 Len=0 MSS 206.71.17.21 192.168.0.7 102.168.0.7 115.1 Sit [Linet Hell] Cartificate 44</td> <td>> Internet Protocol Version 4, Src: 206.71.17.21, Dst: 192.168.0.7</td>	Store Denter Detect TCP Sock Carget Langet Inc Soc. 192.168.0.7 206.71.17.21 192.168.0.7 206.71.17.21 1125.1 60.413.4 49170 443 [SVN] Seq=0 ACK=1 Win=32768 Len=0 MSS=1460 SACK Soc. 192.168.0.7 206.71.17.21 1125.1 Sit [Linet Hell] Sit [Linet Hell] Seq=1 Ack=1 Win=33028 Len=0 MSS 206.71.17.21 192.168.0.7 102.168.0.7 115.1 Sit [Linet Hell] Cartificate 44	> Internet Protocol Version 4, Src: 206.71.17.21, Dst: 192.168.0.7
Bit RM Control Contro Control Control	Store Store <th< td=""><td>> Frame 2: op bytes on wire (480 bits), op bytes captured (480 bits) on inte > Ethernet II, Src: Netgear_6f:d3:9a (78:d2:94:6f:d3:9a), Dst: TonyangN 31:</td></th<>	> Frame 2: op bytes on wire (480 bits), op bytes captured (480 bits) on inte > Ethernet II, Src: Netgear_6f:d3:9a (78:d2:94:6f:d3:9a), Dst: TonyangN 31:
Bit Work Go Chur Angle Static Talghory Wirels Tock Heip Market Static Talghory Wirels Tock Heip Intervention Social Static Talghory Wirels Tock Heip Market Static Talghory Wirels Tock Heip Intervention Social Static Talghory Wirels Tock Heip Market Static Talghory Wirels Tock Heip Intervention Social Static Talghory Wirels Tock Market Static Talghory Wirels Tock Market Static Talghory Wirels Tock Intervention Team Social Static Talghory Wirels Tock Heip Market Static Talghory Wirels Tock Market Minestatic Static Talghory Wirels Tock Intervention Team Social Static Talghory Wirels Tock Heip Market Minestatic Static Talghory Wirels Tock Market Minestatic Static Talghory Mirels Tock Intervention Team Social Static Talghory Mirels Tock Market Minestatic Talghory Mirels Tock Market Minestatic Talghory Mirels Tock Intervention Team Social Static Talghory Mirels Tock Market Minestatic Talghory Mirels Tock Market Minestatic Talghory Mirels Tock Intervention Team Team Team Team Market Minestatic Talghory Mirels Tock Market Minestatic Talghory Mirels Tock Market Minestatic Talghory Mirels Tock Intervention Team Team Team <thteam< th=""> Team Team</thteam<>	Control Defender Defender Fender F	
Bit Entry Openator Spatici Telephony Windex Due Interview Intervie	Carbon Detected Percod Larget Interpret B00 192.168 $206.71.17.21$ 192.168.0.7 206.71.17.21 17CP 62.49170 443 SVN Seq=0 Ack Min=33768 Len=0 MSS=1460 SACK S50 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60.49170 443 SVN Seq=0 Ack=1 Min=33028 Len=0 MSS S62 192.168.0.7 206.71.17.21 TCP 1514 443 49170 [PSH, ACK] Seq=1 Ack=308 Min=35381 Len=14 392 206.71.17.21 192.168.0.7 TCP 1514 443 49170 [PSH, ACK] Seq=1461 Ack=308 Min=35381 Len=14 392 206.71.17.21 192.168.0.7 TLSV1.2 392 Server Hello Seq=348 Ack=308 Min=35381 Len=0 41. 206.71.17.21 192.168.0.7 TLSV1.2 395 Server Hello Corver Key Eape368 Ack=	L 21 1.1581 192.168.0.7 206.71.17.21 TCP 60 4917
Bit Edit Yung Gorgen Ander Geboorg Wieles Teck Help Seat Seattic Technong Wieles Teck Help Seat Seattic Technong Wieles Teck Help Intervent Ferre Seat Seattic Technong Wieles Seattic Technong Seattic Te	ary between the second of the second	20 1.1580 206.71.17.21 192.168.0.7 TCP 60 443 -
Effe full War 6 Cap War Factor Statustic Telephory Wirefe Lock Her Implementation Same	Curr Descent Proced Largeb Inc Source Descent TCP 60 1971 $4rgeb$ Inc Source 2066.71.17.21 TCP 60 443 SYNJ Seq=0 $kin=32768$ Len=0 MSS Source 192.168.0.7 206.71.17.21 TCP 60 443 + 49170 FXNJ Seq=0 Ack=1 Win=8190 Len=0 MSS Source 192.168.0.7 206.71.17.21 TCP 60 49170 FXNJ Seq=1 Ack=1 Win=33028 Len=0 MSS 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 PSH, ACK] Seq=1 Ack=308 Win=35381 Len=14 30. 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 PSH, ACK] Seq=1 Ack=308 Win=35381 Len=14 31. 192.168.0.7 TCP TCP 1514 443 + 49170 PSH, ACK] Seq=1 Ack=308 Win=35381 Len=0	19 1.1019 206.71.17.21 192.168.0.7 TCP 60 443 -
Effe fue Gener Analyze Statistic Telephory Wieles Lock Here Improve Jear Server	Soure Defination Soure Defination Perdocal Largeh Infe Soure Defination Perdocal Largeh Infe Soure Constraint TCP 62 249170 ± 443 [SYM] Seq=0 Min=32768 Len=0 MSS=1460 SACK Soure 192.168.0.7 206.71.17.21 TCP 60 443 ± 9170 [SYM] Seq=0 Ack=1 Win=33028 Len=0 MSS Soure 206.71.17.21 192.168.0.7 Con TCP 1514 443 ± 49170 [SSH, ACK] Seq=1 Ack=1 Win=33028 Len=0 Soure 206.71.17.21 192.168.0.7 TCP 1514 443 ± 49170 [SSH, ACK] Seq=1 Ack=308 Win=35381 Len=14 30. 206.71.17.21 192.168.0.7 TCP 1514 443 ± 49170 [SSH, ACK] Seq=1461 Ack=308 Win=35381 Len=14 30. 206.71.17.21 192.168.0.7 TCP <td>18 1.0825 192.168.0.7 206.71.17.21 TCP 60 49170</td>	18 1.0825 192.168.0.7 206.71.17.21 TCP 60 49170
Eff Eff Yendes Senticit Treeform Winders Total	Source Index Source I	17 1.0793 192.168.0.7 206.71.17.21 TLSV1.2 85 Encry
Eff. Eff. Yew Go Cupher Munders Tore Tore Sector Image of the sector Image of advances Tere Sector Tere Sector Tere Sector Tere Sector	Source Destination Pedacod Largeh Info 90. 192.168.0.7 206.71.17.21 TCP 62 49170 443 [SVN] Seq=0 Min=32768 Len=0 MSS=1460 SACK 90. 192.168.0.7 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 43 \Rightarrow 49170 443 [ACK] Seq=0 Ack=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 1514 443 \Rightarrow 49170 PSH, AcK] Seq=0 Ack=308 Win=33028 Len=0 MSS 57 206.71.17.21 192.168.0.7 TCP 1514 443 \Rightarrow 49170 PSH, AcK] Seq=1 Ack=308 Win=35381 Len=14 39 206.71.17.21 192.168.0.7 TCP 1514 443 \Rightarrow 49170 PSH, AcK] Seq=1461 Ack=308 Win=35381 Len=14 39 206.71.17.21 TCP 1154 443 \Rightarrow <	16 1.0214 206.71.17.21 192.168.0.7 TLSv1.2 221 Appl:
Eff Eff Yew for Go plane Analyse Statistic Telephony Wireles Low Telephony Wey for Control Description Wey for Control Description Wey for Control Description Wey for Control Description Mey for Mey	Surve Dedication Produced Image: Colspan="2">Network Source Dedication Produced Largeh Inde Source Dedication CP Produced Largeh Inde Source Dedication CP Go 2 Surve 443 SYN Seq=0 Win=32768 Len=0 MSS=1460 SACK Source 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 49170 443 [SYN] Seq=0 Ack=1 Min=33028 Len=0 MSS Source 206.71.17.21 192.168.0.7 206.71.17.21 TCP 5114 443 49170 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=0 206.71.17.21 192.168.0.7 TCP 1514 443 49170 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=14 206.71.17.21 192.168.0.7 TIL TIL 392 Server Hello, Certificate 44.1 206.71.17.21	15 0.6909 206.71.17.21 192.168.0.7 TCP 60 443 -
Eff Eff Yiew Go puture Analyse Statistic Telephony Wrieks Tool Hei Image: A decay Terme Sume Sume Control Terme Sume Sum Sum Sum Sum	Surve Destination Protocol term Source Destination Protocol Largh Info Bone 192.168.0.7 206.71.17.21 TCP 62 29170 443 SYN Seq=0 Min=32768 Len=0 MSS Bone 192.168.0.7 206.71.17.21 TCP 60 433 49170 SYN Seq=0 Ack=1 Min=33028 Len=0 MSS S6. 192.168.0.7 206.71.17.21 TCP 60 49170 FM Ack Seq=1 Ack=1 Min=33028 Len=0 MSS S6. 192.168.0.7 206.71.17.21 TISV1.2 361 Client Hello S7. 206.71.17.21 192.168.0.7 TCP 1514 443 49170 [PSH, ACK] Seq=1 Ack=308 Min=35381 Len=0 S7. 206.71.17.21 192.168.0.7 TILSV1.2 392 Server Hello Certificate S7. 206.71.17.21	14 0.6603 192.168.0.7 206.71.17.21 TLSV1.2 731 Appl:
Efic Life Septence Analyze Statistics Telephony Wireless Look Image: Network Network Telephony Source Source Description Performation	Curry Source Ceditation Pedcocd Langth Indo 60 192.168.0.7 206.71.17.21 TCP 62 49170 443 SYNJ Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 49170 SYNJ Seq=0 Min=32768 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 443 [ACK] Seq=0 Ack=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TLSV1.2 361 Client Hello 27 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=14 39 206.71.17.21 192.168.0.7 TLSV1.2 392 Server Hello Len 41 206.71.17.21 192.168.0.7 TLSV1.2 392 Server Hello, Certificate	13 0.5783 206.71.17.21 192.168.0.7 TLSV1.2 105 Chan
Efit Yew Go Gapture Analyze Statistics Telephony Wireles Jools Help Month Tase Source Destination TCP GO 49170 54431 SYNIJ Sequence Main Secure Destination TCP 62 49170 4431 SYNIJ Sequence Main Secure Destination TCP 60 4433 SYNIJ Sequence Ack< Min=33768 Len=0 MSS=1460 SACK 2 0.0255 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 413 + 49170 SYNI, Seq=0 Ack=1 Min=33028 Len=0 MSS 4 0.2068 192.168.0.7 206.71.17.21 192.168.0.7 TCP 50 231.17.21 192.168.0.7 TCP 30 60 433 + 49170 [PSH, ACK] Seq=1 Ack=308 Min=35381 Len=14 6 0.2339 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 [PSH, ACK] Seq=1461 Ack=308 Min=35381 Len=14 7 0.2341 <td>Control Source Centination Centination Protocol Langth Indo Box 192.168.0.7 206.71.17.21 TCP 62 49170 + 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK 56 192.168.0.7 206.71.17.21 TCP 60 443 + 49170 [SYN] Seq=0 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=0 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TLSV1.2 361 Client Hello 57 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 [PSH, ACK] Seq=1 AcK=308 Win=335381 Len=14 59 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 [PSH, ACK] Seq=1 AcK=308 Win=35381 Len=14 59 206.71.17.21 192.168.0.7 TLSV1.2 392 Server</td> <td>12 0.5678 206.71.17.21 192.168.0.7 TCP 60 443 -</td>	Control Source Centination Centination Protocol Langth Indo Box 192.168.0.7 206.71.17.21 TCP 62 49170 + 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK 56 192.168.0.7 206.71.17.21 TCP 60 443 + 49170 [SYN] Seq=0 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=0 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TLSV1.2 361 Client Hello 57 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 [PSH, ACK] Seq=1 AcK=308 Win=335381 Len=14 59 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 [PSH, ACK] Seq=1 AcK=308 Win=35381 Len=14 59 206.71.17.21 192.168.0.7 TLSV1.2 392 Server	12 0.5678 206.71.17.21 192.168.0.7 TCP 60 443 -
File Edit Yuev Sutvicts Telephony Wineles Job Image: Source Sour	Control Source Centination Pedicod Langth Indo 800. 192.168.0.7 206.71.17.21 TCP 62 49170 443 SYNJ Seq=0 Min=32768 Len=0 MSS=1460 SACK 55. 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 49170 SVNJ Seq=0 Min=32768 Len=0 MSS 56. 192.168.0.7 206.71.17.21 TCP 60 49170 443 [ACK] Seq=0 AcK=1 Win=33028 Len=0 MSS 56. 192.168.0.7 206.71.17.21 TCP 150 443 49170 [PSH, ACK] Seq=1 AcK=1 Win=33028 Len=0 MSS 57. 206.71.17.21 192.168.0.7 TLSV1.2 361 Client Hello Seq=1 AcK=1 Win=33028 Len=14 59. 206.71.17.21 192.168.0.7 TCP 1514 443 49170 [PSH, ACK] Seq=1 AcK=308 <t< td=""><td>11 0.5446 192.168.0.7 206.71.17.21 TLSV1.2 180 Clien</td></t<>	11 0.5446 192.168.0.7 206.71.17.21 TLSV1.2 180 Clien
File Edit Yuev G Capture Analyze Statistic Telephony Wineles Job Image: Analyze Set is in telephony Image: Analyze	Control Source Centination Pedicod Langth Indo 800. 192.168.0.7 206.71.17.21 TCP 62 49170 + 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 443 + 49170 [SYN] Seq=0 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=0 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TLSV1.2 361 Client Hello 77 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 [PSH, ACK] Seq=1 AcK=308 Win=35381 Len=14 39 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 [PSH, ACK] Seq=1461 AcK=308 Win=35381 Len=14 39 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170	10 0.2438 192.168.0.7 206.71.17.21 TCP 60 49170
File Edit Yuev G Capture Analyze Statistics Telephony Wireles Job Image: Statistic Image: Statistics Image: Statistics <td< td=""><td>Control Source Centination Pedacod Langth Indo 800. 192.168.0.7 206.71.17.21 TCP 62 49170 443 SYNJ Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 443 49170 SYNJ Seq=0 Min=32768 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=1 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=1 AcK=1 Win=33028 Len=0 MSS 57 206.71.17.21 192.168.0.7 TLSV1.2 361 Client Hello 27 206.71.17.21 192.168.0.7 TCP 1514 443 49170 [PSH, AcK] Seq=1 AcK=308 Win=35381 Len=14 39 206.71.17.21 192.168.0.7 TCP</td><td>9 0.2435 206.71.17.21 192.168.0.7 TLSV1.2 396 Serve</td></td<>	Control Source Centination Pedacod Langth Indo 800. 192.168.0.7 206.71.17.21 TCP 62 49170 443 SYNJ Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 443 49170 SYNJ Seq=0 Min=32768 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=1 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=1 AcK=1 Win=33028 Len=0 MSS 57 206.71.17.21 192.168.0.7 TLSV1.2 361 Client Hello 27 206.71.17.21 192.168.0.7 TCP 1514 443 49170 [PSH, AcK] Seq=1 AcK=308 Win=35381 Len=14 39 206.71.17.21 192.168.0.7 TCP	9 0.2435 206.71.17.21 192.168.0.7 TLSV1.2 396 Serve
File Edit Yuev G Capture Analyze Satistics Telephony Wineles Tool Image Image	Control Source Centination Centination Protocol Langth Info 800. 192.168.0.7 206.71.17.21 TCP 62 49170 443 SYNJ Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 443 49170 SVN, ACKJ Seq=0 Ack=1 Win=83028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 150 443 49170 [SVN] Seq=1 Ack=1 Win=33028 Len=0 MSS 57 206.71.17.21 192.168.0.7 TLSV1.2 361 Client Hello 27 206.71.17.21 192.168.0.7 TCP 1514 443 49170 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=14 39 206.71.17.21	8 0.2346 192.168.0.7 206.71.17.21 TCP 60 49170
File Edit Yuev G Capture Analyze Statistic Telephony Wineles Tools Help A poly a dopoly R and a dopoly <td>Control Source Centination Centination Produced Langth Info 800. 192.168.0.7 206.71.17.21 TCP 62 49170 + 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 443 4 9170 SVN, ACK] Seq=0 Ack=1 Win=83028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=1 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=1 AcK=1 Win=33028 Len=0 57 192.168.0.7 206.71.17.21 TLSV1.2 361 Client Hello 27 206.71.17.21 192.168.0.7 TCP 1514 443 49170 [PSH, AcK] Seq=1 Ack=308 Win=35381 Len=143 39 <td< td=""><td>7 0.2341 206.71.17.21 192.168.0.7 TLSV1.2 392 Servi</td></td<></td>	Control Source Centination Centination Produced Langth Info 800. 192.168.0.7 206.71.17.21 TCP 62 49170 + 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 443 4 9170 SVN, ACK] Seq=0 Ack=1 Win=83028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=1 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 + 443 [ACK] Seq=1 AcK=1 Win=33028 Len=0 57 192.168.0.7 206.71.17.21 TLSV1.2 361 Client Hello 27 206.71.17.21 192.168.0.7 TCP 1514 443 49170 [PSH, AcK] Seq=1 Ack=308 Win=35381 Len=143 39 <td< td=""><td>7 0.2341 206.71.17.21 192.168.0.7 TLSV1.2 392 Servi</td></td<>	7 0.2341 206.71.17.21 192.168.0.7 TLSV1.2 392 Servi
File Edit Yiew G Capture Analyze Satistics Telephony Wineless I observation No. Tme Satistics Satisti	Carry Source Destination Pedacod Langth Info 80re Destination 206.71.17.21 TCP 62.49170 + 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK 90 192.168.0.7 206.71.17.21 TCP 62.49170 + 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 TCP 60.443 + 49170 [SYN] Seq=0 AcK=1 Win=33028 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60.49170 + 443 [ACK] Seq=1 AcK=1 Win=33028 Len=0 58 192.168.0.7 206.71.17.21 TLSV1.2 361 Client Hello 27 206.71.17.21 192.168.0.7 TCP 1514 443 + 49170 [PSH, AcKE] Seq=1 AcK=308 Win=35381 Len=14	6 0.2339 206.71.17.21 192.168.0.7 TCP 1514 443 -
File Edit Yuew Go Capture Analyze Satistics Telephony Wineless Job No. Time Source Source Destination Protocol Info 1 0.00000 192.1688.0.7 206.71.17.21 Protocol Length Info 2 0.0255 206.71.17.21 192.168.0.7 206.71.17.21 TCP 60 49170 443 SVNJ Seq=0 Ack=1 Win=8300 Len=0 MSS 3 0.0255 192.168.0.7 206.71.17.21 TCP 60 49170 443 SVNJ Seq=0 Ack=1 Win=33028 Len=0 MSS 4 0.20688 192.168.0.7 206.71.17.21 TCP 60 49170 443 Seq=1 Ack=1 Win=33028 Len=0 MSS 4 0.20688 192.168.0.7 206.71.17.21 TCP 60 49170 443 Ack=1 Win=33028 Len=0 4 0.20688 192.168.0.7 206.71.17.21 TLSV1.2 361 Client Hello	Lamph Lamph Info Source Destination Pedicod Langth Info 80 192.168.0.7 206.71.17.21 TCP 62 49170 > 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 TCP 60 443 49170 [SYN] Seq=0 Ack=1 Win=8190 Len=0 MSS 56 192.168.0.7 206.71.17.21 TCP 60 49170 > 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0 68 192.168.0.7 206.71.17.21 TLSv1.2 361 Client Hello	5 0.2327 206.71.17.21 192.168.0.7 TCP 1514 443 -
File Edit Yiew G Captva Analyze Satistics Telephony. Wineless Loop Analyze Satistics Telephony. Wineless Loop Analyze Satistics Satistics Telephony. Wineless Loop Analyze Satistics	Contribution Source Destination Bounce Centination 00 192.168.0.7 206.71.17.21 206.71.17.21 192.168.0.7 206.71.17.21 192.168.0.7 206.71.17.21 192.168.0.7 206.71.17.21 192.168.0.7 TCP 60.43 + 49170 55 192.168.0.7 192.168.0.7 TCP 60.43 + 49170 56 192.168.0.7	4 0.2068 192.168.0.7 206.71.17.21 TLSv1.2 361 Clie
File Edit Yiew Go Capture Analyze Satistics Telephony Wineless Loop Help A poly a dopuy file - Conf/- Ima	Code/s Source Deskination Protocol Langth Info Source Source Deskination Protocol Info 00 192.168.0.7 206.71.17.21 TCP 62 49170 + 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK 55 206.71.17.21 192.168.0.7 TCP 60 443 + 49170 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS	3 0.0256 192.168.0.7 206.71.17.21 TCP 60 49170
File Edit Yiew Go Capture Analyze Satistics Telephony Wineless Loob Help A land Image	COM/> Source Deskination Protocol Length Info Source 206.71.17.21 TCP 62 49170 + 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK	2 0.0255 206.71.17.21 192.168.0.7 TCP 60 443 -
File Edit Yiew Go Capture Analyze Satistics Telephony Wireless Loois Help Image:	Cotr/>> Source Deskination Protocol Length	1 0.0000 192.168.0.7 206.71.17.21 TCP 62 49170
File Edit View Go Capture Analyze Statistics Telephony Wireless Iools Help Image: Source Statistics Telephony Wireless Iools Help Image: Source Statistics Telephony Help		No. Time Source Destination Protocol Length Info
Eile Edit View <u>Go</u> Capture Analyze Statistics Telephony Wireless Iools Help M B M O I B N C I Q e I E E E E E E E E E E		Apply a display filter < Chri-/>
Eile Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help		
	Gapture Analyze Statistics Telephony. Wireless Iools Help	<u>File Edit View Go Capture Analyze Statistics Telephony Wireless Iools Help</u>

Appendix B Successful Withdrawal \$60

Figure 11. Successful Withdrawal of \$60 (Wireshark)

File Home Inst	ert Page Layout	Formulas	Data	Review View Q Tell me what you want to do
Cut	Calibri 🕆 11	× A A		
Paste 🔶 🗸 Format Painter	В / Ц - П -	🖓 - A -		
Clipboard	Font	a		Alignment 🖙 Number 🖙
G28 +	$x \checkmark x$			
B	D	m	T	G
1 No. Time Sourc	Destination	Protocol	Length	Info
2 1 0 192.1	68.0.7 206.71.17.21	L TCP	62	49170 > 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK_PERM=1
				443 > 49170 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0
2 0.02333 200.7 2 3 0.02533 200.7	68 0 7 206 71 17 21		5	19170 > 443 [ACK] Sec-1 Ack-1 Win-32028 Len-0
5 4 0.20683 192.1	68.0.7 206.71.17.21	TLSv1.2	361	Client Hello
				443 > 49170 [PSH, ACK] Seq=1 Ack=308 Win=35381
6 5 0.23275 206.7	1.17.21 192.168.0.7	TCP	1514	Len=1460 [TCP segment of a reassembled PDU]
7 6 0.23398 206.7	1.17.21 192.168.0.7	TCP	1514	443 > 49170 [PSH, ACN] SEQ=1401 ACK=300 WITE33301 Len=1460 [TCP segment of a reassembled PDU]
8 7 0.23417 206.7	1.17.21 192.168.0.7	TLSv1.2	392	Server Hello, Certificate
9 8 0.23469 192.1	68.0.7 206.71.17.21	TCP	60	49170 > 443 [ACK] Seq=308 Ack=2921 Win=33028 Len=0
10 9 0.24351 206.7	1.17.21 192.168.0.7	TLSv1.2	396	Server Key Exchange, Server Hello Done
11 10 0.24381 192.1	68.0.7 206.71.17.21	L TCP	60	49170 > 443 [ACK] Seq=308 Ack=3601 Win=32348 Len=0
12 11 0.54464 192.1	68.0.7 206.71.17.21	TLSv1.2	180	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
13 12 0.5679 206.7	1.17.21 192.168.0.7	TCP	60	443 > 49170 [ACK] Seq=3601 Ack=434 Win=35255 Len=0
14 13 0.57839 206.7	1.17.21 192.168.0.7	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message
15 14 0.66037 192.1	68.0.7 206.71.17.21	L TLSv1.2	731	Application Data
16 15 0.69094 206.7	1.17.21 192.168.0.7	TCP	60	443 > 49170 [ACK] Seq=3652 Ack=1111 Win=39848 Len=0
17 16 1.02141 206.7	1.17.21 192.168.0.7	TLSv1.2	221	Application Data
18 17 1.07933 192.1	68.0.7 206.71.17.21	L TLSv1.2	85	Encrypted Alert
19 18 1.08253 192.1	68.0.7 206.71.17.21	L TCP	60	49170 > 443 [HN, ACK] Seq=1142 Ack=3819 Win=32130 Len=0
20 19 1.10198 206.7	1.17.21 192.168.0.7	ТСР	60	443 > 49170 [ACK] Seq=3819 Ack=1142 Win=39817 Len=0
21 20 1.15803 206.7	1.17.21 192.168.0.7	ТСР	60	443 > 49170 [FIN, ACK] Seq=3819 Ack=1143 Win=39817 Len=0
22 21 1.15814 192.1	68.0.7 206.71.17.21	L TCP	60	49170 > 443 [ACK] Seq=1143 Ack=3820 Win=32130 Len=0
23 24				
26				
27			Ļ	

Figure 12. Successful Withdrawal of \$60 (CSV)

C	0	\sim	\sim	~ ~																			_		$\left(\right)$	Ņ,	2	Elle	
Balance Check Succes	000 00 a0	Transmission (Internet Proto	Frame 2: 60 by Ethernet II, S	21 2.3332	20 2.3329	19 2.2782	18 2.2596	17 2.2563	16 2.0813	15 1.9646	14 1.4291	13 1.4062	12 1.1969	11 1.1862	10 1.1539	9 0.2800	8 0.2794	7 0.2790	6 0.2787	5 0.2775	4 0.2069	3 0.0354	2 0.0353	1 0.0000	Time	Apply a display filter <ctrl-></ctrl->	<u>E</u> dit <u>V</u> iew <u>G</u> o <u>C</u> a _l	3alance Check Successful F
sful Filtered.pcapng	7c 31 5d 37	Control Protoco	col Version 4,	rtes on wire (4 Src: Netgear_6f	192.168.0.7	206.71.17.21	206.71.17.21	192.168.0.7	192.168.0.7	192.168.0.7	206.71.17.21	206.71.17.21	192.168.0.7	206.71.17.21	206.71.17.21	192.168.0.7	192.168.0.7	206.71.17.21	206.71.17.21	206.71.17.21	206.71.17.21	192.168.0.7	192.168.0.7	206.71.17.21	192.168.0.7	Source		pture <u>A</u> nalyze <u>S</u> tatistic:	iltered.pcapng
	78 d2 94 6f d3 9	l, Src Port: 443, Dst Po	Src: 206.71.17.21, Dst:	80 bits), 60 bytes captu :d3:9a (78:d2:94:6f:d3:9	206.71.17.21	192.168.0.7	192.168.0.7	206.71.17.21	206.71.17.21	206.71.17.21	192.168.0.7	192.168.0.7	206.71.17.21	192.168.0.7	192.168.0.7	206.71.17.21	206.71.17.21	192.168.0.7	192.168.0.7	192.168.0.7	192.168.0.7	206.71.17.21	206.71.17.21	192.168.0.7	206.71.17.21	Destination		: Telephony <u>W</u> ireless <u>T</u> ools <u>H</u> elp	
	a 08 00 4	rt: 49166, S	192.168.0.7	red (480 bit a), Dst: Tor	TCP	TCP	ТСР	TCP	TLSv1.2	TCP	TLSv1.2	TCP	TLSv1.2	TLSv1.2	TCP	TLSv1.2	TCP	TLSv1.2	TLSv1.2	TCP	TCP	TLSv1.2	TCP	TCP	TCP	Protocol			
Packets: 21 - Displayed: 21 (100.0%)		Seq: 0, Ack: 1, Len: 0		ts) on interface 0 nyangN 31:5d:37 (00:a0:7c:31:5d:37)	60 49166 → 443 [ACK] Seq=1143 Ack=3820 Win=32130 Len=0	60 443 → 49166 [FIN, ACK] Seq=3819 Ack=1143 Win=39817 Len=0	60 443 → 49166 [ACK] Seq=3819 Ack=1142 Win=39817 Len=0	60 49166 → 443 [FIN, ACK] Seq=1142 Ack=3819 Win=32130 Len=0	85 Encrypted Alert	60 49166 → 443 [ACK] Seq=1111 Ack=3819 Win=32130 Len=0	221 Application Data	60 443 → 49166 [ACK] Seq=3652 Ack=1111 Win=39848 Len=0	731 Application Data	105 Change Cipher Spec, Encrypted Handshake Message	60 443 → 49166 [ACK] Seq=3601 Ack=434 Win=35255 Len=0	180 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message	60 49166 → 443 [ACK] Seq=308 Ack=2921 Win=33028 Len=0	396 Server Key Exchange, Server Hello Done	392 Server Hello, Certificate	1514 443 → 49166 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TCP segme	1514 443 → 49166 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460 [TCP segment	361 Client Hello	60 49166 → 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0	60 443 → 49166 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1436	62 49166 → 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK_PERM=1	Length Info			

Appendix C Balance Check Successful Zero Balance

Figure 13. Balance Check Successful Zero Balance (Wireshark)

le Home Insert Page La	yout Formulas	Data	Review	View 🛛 Tell me what yc;; want to do
X Cut Calibri	→ 11 → A	A A	≡ <mark>×</mark>	P Wrap Text General ▼ P Vrap Text General Vormal
re vopy ·	•	• 		Herge & Center * \$ * % * 60 .00 Conditional Format as Neutral
Clipboard	Font	Ð	Alig	nment ra Number ra
7 - I X - I	fx			
A B C	D	m	Ŧ	6
No. Time Source	Destination	Protocol	Length	Info
1 0 192.168.0.7	206.71.17.21	TCP	62	49166 > 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 SACK_PERM=1
2 0.035342 206.71.17.21	192.168.0.7	TCP	60	443 > 49166 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1436
3 0.035477 192.168.0.7	206.71.17.21	TCP	60	49166 > 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0
4 0.206987 192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
5 0.277568 206.71.17.21	192.168.0.7	TCP	1514	443 > 49166 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460 [TCP segment of a reassembled PDU]
				443 > 49166 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460
6 0.278768 206.71.17.21	192.168.0.7	TCP	1514	[TCP segment of a reassembled PDU]
8 0.32033 206.71.17.21	192.168.0.7	TISV1.2	305	Server Key Evchange Server Hello Done
9 0.280002 192.168.0.7	206.71.17.21	TCP	60	49166 > 443 [ACK] Seq=308 Ack=2921 Win=33028 Len=0
				Client Key Exchange, Change Cipher Spec, Encrypted Handshake
10 1.153921 192.168.0.7	206.71.17.21	TLSv1.2	180	Message
11 1.186262 206.71.17.21	192.168.0.7	TCP	60	443 > 49166 [ACK] Seq=3601 Ack=434 Win=35255 Len=0
12 1.196955 206.71.17.21	192.168.0.7	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message
13 1.406209 192.168.0.7	206.71.17.21	TLSv1.2	731	Application Data
14 1.429117 206.71.17.21	192.168.0.7	TCP	60	443 > 49166 [ACK] Seq=3652 Ack=1111 Win=39848 Len=0
15 1.964645 206.71.17.21	192.168.0.7	TLSv1.2	221	Application Data
16 2.0813 192.168.0.7	206.71.17.21	TCP	60	49166 > 443 [ACK] Seq=1111 Ack=3819 Win=32130 Len=0
18 2 259631 192 168 0 7	200.71.17.21	TCP	60 00	Encrypted Alert 49166 > 443 [FIN ACK] Sect=1142 Ack=3819 Win=32130 Len=0
19 2.278283 206.71.17.21	192.168.0.7	TCP	60	443 > 49166 [ACK] Seq=3819 Ack=1142 Win=39817 Len=0
20 2.332919 206.71.17.21	192.168.0.7	TCP	60	443 > 49166 [FIN, ACK] Seq=3819 Ack=1143 Win=39817 Len=0
21 2.333214 192.168.0.7	206.71.17.21	TCP	60	49166 > 443 [ACK] Seq=1143 Ack=3820 Win=32130 Len=0

Figure 14. Balance Check Successful Zero Balance (CSV)

File 8 Insuffiencent Funds Filtered.pcapng Ethernet II, Src: TonyangN_31:5d:37 (00:a0:7c:31:5d:37), Dst: Netgear_6f:d3:9a (78:d2:94:6f:d3:9a) 0000 N Transmission Control Protocol, Src Port: 49167, Dst Port: 443, Seq: 0, Internet Protocol Version 4, Src: 192.168.0.7, Dst: 206.71.17.21 Frame 3: 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on interface 0 <u>E</u>dit <u>V</u>iew Insuffiencent Funds Filtered.pcapng 24 2.3564... 23 2.3561... 206.71.17.21 22 2.2892... 21 2.2776... 20 2.2215... 19 1.8700... 18 1.7152... 17 1.2377... 16 1.2124... 15 1.1485... 14 1.1485... 13 1.0645... 12 0.4142... 11 0.4140... 10 0.4050... 90.4045... 80.4041... 6 0.3764.... 7 0.4030... 5 0.0746... 2 0.0363... 10.0000... 4 0.0743... 3 0.0469.. ۲ 78 d2 94 6f į, <u>G</u>o <u>C</u>apture <u>A</u>nalyze 192.168.0.7 206.71.17.21 192.168.0.7 192.168.0.7 192.168.0.7 192.168.0.7 192.168.0.7 192.168.0.1 192.168.0.7 206.71.17.21 206.71.17.21 206.71.17.21 206.71.17.21 192.168.0.7 206.71.17.21 192.168.0.7 206.71.17.21 206.71.17.21 206.71.17.21 192.168.0.7 192.168.0.7 206.71.17.21 192.168.0.7 Source d S Statistics 9a Telephony Wireless Iools Help 00 192.168.0.7 206.71.17.21 206.71.17.21 192.168.0.7 206.71.17.21 206.71.17.21 192.168.0.7 192.168.0.7 206.71.17.21 192.168.0.7 192.168.0.7 206.71.17.21 206.71.17.21 192.168.0.7 206.71.17.21 192.168.0.7 192.168.0.7 206.71.17.21 206.71.17.21 192.168.0.7 206.71.17.2 192.168.0.7 192.168.0.7 192.168.0.1 aØ 7c μ ъ 37 80 Protocol DNS DNS TCP TCP TCP TLSV1.2 TCP TCP TCP TCP TLSv1.2 TCP TLSv1.2 TCP TLSv1.2 TLSv1.2 TCP TLSv1.2 TCP TLSv1.2 TCP TLSv1.2 TCP 00 45 1514 443 → 49167 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TCP seg 1514 443 → 49167 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460 [TCP segmen Length 180 Client Key Exchange, Change Cipher Spec, Encrypted Handshake Messag 396 Server Key Exchange, Server Hello Done 392 Server Hello, Certificate 361 Client Hello 221 Application Data 731 Application Data 105 Change Cipher Spec, Encrypted Handshake Message 00 60 49167 + 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0 97 Standard query response 0x03da A EFTDEBITATM.FNFIS.COM A 206.71.17. 81 Standard query 0x03da A EFTDEBITATM.FNFIS.COM 60 49167 + 443 [ACK] 60 443 → 49167 [FIN, ACK] Seq=3819 Ack=1143 Win=39817 Len=0 60 49167 → 443 [FIN, ACK] 60 443 → 49167 [ACK] Seq=3819 Ack=1142 Win=39817 Len=0 85 Encrypted Alert 60 49167 → 443 [ACK] Seq=1111 Ack=3819 Win=32130 Len=0 60 443 → 49167 [ACK] Seq=3652 Ack=1111 Win=39848 Len=0 60 443 → 49167 [ACK] Seq=3601 Ack=434 Win=35255 Len=0 60 49167 + 443 [ACK] Seq=308 Ack=3601 Win=32348 Len=0 60 49167 + 443 [ACK] Seq=308 Ack=2921 Win=33028 Len=0 60 443 → 49167 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1436 $62\ 49167\ o\ 443$ Info Len: 9 [NAS Seq=1143 Seq=0 Win=32768 Seq=1142 Ack=3819 Win=32130 Len=0 Ack=3820 Win=32130 Len=0 Len=0 MSS=1460 SACK_PERM=1 Packets: 24 · Displayed: 24 (100.0%)

Appendix D Insufficient Funds

Figure 15. Insufficient Funds (Wireshark)

9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9	3 ins Funds - Exc	cel
File Home Insert Page Layout Formulas Data Review View Q Tell me what	what yco want to do	
A Cut Calibri - 12 - A A ≡ ≡ ≫ - ■ Wrap Text	General -	Normal
Paste "≧ Copy • B I <u>U</u> • ⊡ • ☆ • ▲ • ≡ ≡ ≣ Merge & Center •	ter - \$ - % + *0 *0 Conditional Format a Formatting - Table -	as Neutral
Clipboard 13 Font 13 Alignment 13	ra Number ra	tts
G_{24} \rightarrow $: \times \checkmark f_{k} $		
A B C D E F	G	I
1 No. Time Source Destination Protocol Length Info		
2 1 0.046082 102 168 0.7 206 71 17 21 TCP 62 SACK DERM-1	N] Seq=0 Win=32768 Len=0 MSS=1460	
443 > 49167 [SYN. AC	V. ACKI Seq=0 Ack=1 Win=8190 Len=0	
3 2 0.074392 206.71.17.21 192.168.0.7 TCP 60 MSS=1436		
4 3 0.07462 192.168.0.7 206.71.17.21 TCP 60 49167 > 443 [ACK] See	K] Seq=1 Ack=1 Win=33028 Len=0	
5 4 0.376401 192.168.0.7 206.71.17.21 TLSv1.2 361 Client Hello		
6 5 0.403043 206.71.17.21 192.168.0.7 TCP 1514 [TCP segment of a reas	H, ACK] Seq=1 Ack=308 Win=35381 Len=146 3 reassembled PDU]	50
7 6 0.404197 206.71.17.21 192.168.0.7 TCP 1514 Len=1460 [TCP segmen	H, ACK] Seq=1461 Ack=308 Win=35381 gment of a reassembled PDU]	
8 7 0.404573 206.71.17.21 192.168.0.7 TLSv1.2 392 Server Hello, Certificat	ificate	
9 8 0.405047 192.168.0.7 206.71.17.21 TCP 60 49167 > 443 [ACK] See	K] Seq=308 Ack=2921 Win=33028 Len=0	
10 9 0.414094 206.71.17.21 192.168.0.7 TLSv1.2 396 Server Key Exchange, S	nge, Server Hello Done	
11 10 0,41424 192.168.0.7 206.71.17.21 ICP 60 49167 > 443 JACKJ See	KJ Seq=308 Ack=3601 Win=32348 Len=0	
12 11 1.064564 192.168.0.7 206.71.17.21 TLSv1.2 180 Handshake Message	ige, change cipher open, che provi ige	
13 12 1.148536 206.71.17.21 192.168.0.7 TCP 60 443 > 49167 [ACK] See	K] Seq=3601 Ack=434 Win=35255 Len=0	
14 13 1.148537 206.71.17.21 192.168.0.7 TLSv1.2 105 Change Cipher Spec, E	ec, Encrypted Handshake Message	
15 14 1.212442 192.168.0.7 206.71.17.21 TLSv1.2 731 Application Data		
16 15 1.237744 206.71.17.21 192.168.0.7 TCP 60 443 > 49167 [ACK] See	K] Seq=3652 Ack=1111 Win=39848 Len=0	
17 16 1.715272 206.71.17.21 192.168.0.7 TLSv1.2 221 Application Data		
18 17 1.870099 192.168.0.7 206.71.17.21 TCP 60 49167 > 443 [ACK] Sec	K] Seq=1111 Ack=3819 Win=32130 Len=0	
19 18 2.22151 192.168.0.7 206.71.17.21 TLSv1.2 85 Encrypted Alert		
20 19 2.277644 206.71.17.21 192.168.0.7 TCP 60 443 > 49167 [ACK] See	K] Seq=3819 Ack=1142 Win=39817 Len=0	
21 20 2.289297 192.168.0.7 206.71.17.21 TCP 60 49167 > 443 [FIN, ACK	I, ACK] Seq=1142 Ack=3819 Win=32130 Len-	=0
22 21 2.356185 206.71.17.21 192.168.0.7 TCP 60 443 > 49167 [FIN, ACk	I, ACK] Seq=3819 Ack=1143 Win=39817 Len:	=0
23 22 2.356483 192.168.0.7 206.71.17.21 TCP 60 49167 > 443 [ACK] See	K] Seq=1143 Ack=3820 Win=32130 Len=0	
24 23		
3 Ins Funds +		
Ready		

Figure 16. Insufficient Funds (CSV)

~ 0 2 Fe Balance Check Incorrect PIN Filtered.pcapng 0000 Transmission Control Protocol, Src Port: 443, Dst Port: 49165, Seq: 0, Ack: 1, Len: 0 Ethernet II, Src: Netgear_6f:d3:9a (78:d2:94:6f:d3:9a), Dst: TonyangN_31:5d:37 (00:a0:7c:31:5d:37) Internet Protocol Version 4, Src: 206.71.17.21, Dst: 192.168.0.7 Frame 2: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0 Apply a display filter <u>E</u>dit <u>V</u>iew Balance Check Incorrect PIN Filtered.p 22 1.8260... 192.168.0.7 21 1.8258... 20 1.8013... 19 1.7889... 18 1.7103... 17 1.6395.... 15 1.0034... 14 0.9800... 13 0.9454... 16 1.5069... 12 0.9454.. 11 0.9059.. 10 0.4269... 9 0.4266... 80.4145.. 6 0.4137... 5 0.4125... 4 0.3812... 30.1118... 2 0.1039... 7 0.4140... 1 0.0000.. ۲ 00 Time <u>Go</u> <u>C</u>apture <u>A</u>nalyze <u>Statistics</u> Telephony <u>W</u>ireless <u>I</u>ools <u>H</u>elp aØ 7c × 206.71.17.21 206.71.17.21 206.71.17.21 192.168.0.7 192.168.0.7 192.168.0.7 206.71.17.21 192.168.0.7 206.71.17.21 192.168.0.7 192.168.0.7 206.71.17.21 206.71.17.21 192.168.0.7 206.71.17.21 206.71.17.21 192.168.0.7 192.168.0.7 206.71.17.21 206.71.17.21 192.168.0.7 Source μ ъ 37 87 206.71.17.21 192.168.0.7 206.71.17.21 192.168.0.7 206.71.17.21 206.71.17.21 206.71.17.21 206.71.17.21 206.71.17.21 206.71.17.21 206.71.17.21 206.71.17.21 192.168.0.7 Destinatio 192.168.0.7 192.168.0.7 192.168.0.7 192.168.0.7 192.168.0.7 192.168.0.7 192.168.0.7 206.71.17.2 192.168.0.7 d2 94 бť а 9a 80 TCP TCP TLSv1.2 TLSv1.2 Protoco TCP TCP TCP TCP TCP TLSv1.2 TCP TCP TLSv1.2 TLSv1.2 TLSv1.2 TCP TCP TLSv1.2 TLSv1.2 00 5 1514 443 → 49165 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TC Length $1514\ 443\ +\ 49165$ 105 Change Cipher Spec, Encrypted Handshake Message 180 Client Key Exchange, Change Cipher Spec, Encrypted Handshake M 396 Server Key Exchange, Server Hello Done 392 Server Hello, Certificate 361 Client Hello 80 60 49165 + 443 [FIN, ACK] Seq=1142 Ack=3819 Win=32130 Len=0 60 443 + 49165 [FIN, ACK] Seq=3819 Ack=1143 Win=39817 Len=0 60 49165 + 443 [ACK] Seq=1143 Ack=3820 Win=32130 Len=0 221 Application Data 731 Application Data 60 49165 → 443 [ACK] Seq=308 Ack=2921 Win=33028 Len=0 60 49165 → 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0 60 443 → 49165 60 443 → 49165 [ACK] Seq=3819 Ack=1142 Win=39817 Len=0 85 Encrypted Alert 60 49165 → 443 [ACK] Seq=1111 Ack=3819 Win=32130 Len=0 60 443 → 49165 [ACK] Seq=3652 Ack=1111 Win=39848 Len=0 60 443 → 49165 [ACK] Seq=3601 Ack=434 Win=35255 Len=0 60 49165 + 443 [ACK] Seq=308 Ack=3601 Win=32348 Len=0 $62\ 49165\ o\ 443$ Info [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460 [TCP s [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1436 [NAS] Seq=0 Win=32768 Len=0 MSS=1460 SACK_PERM=1 Packets: 22 · Displayed: 22 (100.0%)

Appendix E Balance Check, Wrong PIN

Figure 17. Balance Check, Wrong Pin (Wireshark)

- 1	ē	Home Insert Page La	ayout Formulas	Data	Review	View Q Tell me what y	o want to do				
		Cut Calibri	~ 11 ~ A	>, 	- 48	Wrap Text	General •	×		Normal	Bad
Pas	in	Format Painter B I U	- - A			Merge & Center *	\$ - % + [∞] 0, 00, 00	Conditional Formatting *	ormat as Table *	Neutral	Calculation
	G	pboard 15	Font	2	Alig	nment 13	Number 1			S	tyles
G3	0	•	fr								
<u>.</u>	Þ	в	D	m	Ŧ			ត			I
-	No.	Time Source	Destination	Protocol	Length	Info					
2	1	0 192.168.0.7	206.71.17.21	TCP	62	49165 > 443 [SYN] S	eq=0 Win=32768 I	en=0 MSS=14	60 SACK	PERM=1	
ω	2	0.103904 206.71.17.21	192.168.0.7	TCP	60	443 > 49165 [SYN, A	CK] Seq=0 Ack=1 V	Vin=8190 Len	=0 MSS=	-1436	
4	ω	0.111887 192.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] S	eq=1 Ack=1 Win=:	33028 Len=0			
S	4	0.381249 192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello					
						443 > 49165 [PSH, A	CK] Seq=1 Ack=30	8 Win=35381	Len=146	0 [TCP segme	nt
6	5	0.41254 206.71.17.21	192.168.0.7	TCP	1514	of a reassembled PD	U				
						443 > 49165 [PSH, A	CK] Seq=1461 Ack	=308 Win=353	81 Len=	:1460 [TCP	
7	6	0.413755 206.71.17.21	192.168.0.7	TCP	1514	segment of a reasser	nbled PDU]				
8	7	0.414071 206.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certific	ate				
9	8	0.414577 192.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] S	eq=308 Ack=2921	Win=33028 Le	en=0		
10	9	0.42667 206.71.17.21	192.168.0.7	TLSv1.2	396	Server Key Exchange	, Server Hello Done	CD.			
11	10	0.426977 192.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] S	eq=308 Ack=3601	Win=32348 Le	en=0		
12	11	0.905932 192.168.0.7	206.71.17.21	TLSv1.2	180	Client Key Exchange,	Change Cipher Sp	ec, Encrypted	Handsh	ake Message	
13	12	0.945464 206.71.17.21	192.168.0.7	TCP	60	443 > 49165 [ACK] S	eq=3601 Ack=434	Win=35255 Le	en=0		
14	13	0.945491 206.71.17.21	192.168.0.7	TLSv1.2	105	Change Cipher Spec,	Encrypted Handsh	nake Message			
15	14	0.98003 192.168.0.7	206.71.17.21	TLSv1.2	731	Application Data					
16	15	1.003492 206.71.17.21	192.168.0.7	TCP	60	443 > 49165 [ACK] S	eq=3652 Ack=111	1 Win=39848	Len=0		
17	16	1.506922 206.71.17.21	192.168.0.7	TLSv1.2	221	Application Data					
18	17	1.639561 192.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] S	eq=1111 Ack=381	9 Win=32130	Len=0		
19	18	1.710325 192.168.0.7	206.71.17.21	TLSv1.2	85	Encrypted Alert					
20	19	1.788997 206.71.17.21	192.168.0.7	TCP	60	443 > 49165 [ACK] S	eq=3819 Ack=114;	2 Win=39817	Len=0		
21	20	1.801319 192.168.0.7	206.71.17.21	TCP	60	49165 > 443 [FIN, A	CK] Seq=1142 Ack=	-3819 Win=32	130 Len:	=0	
22	21	1.825801 206.71.17.21	192.168.0.7	TCP	60	443 > 49165 [FIN, A	CK] Seq=3819 Ack=	-1143 Win=39	817 Len:	=0	
23	22	1.826029 192.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] S	eq=1143 Ack=3820	0 Win=32130	Len=0		
24											
25											
26											
27											
28											
29											
-	-	1 Balance Check Inc	orrect PIN	÷							*
Read	5										
and the second s											

Figure 18. Balance Check, Wrong Pin (CSV)

Appendix F Successful Balance Check \$110

Figure 19. Successful Balance Check \$110 (Wireshark)

Wrap Text General Ceneral Normal Bad Merge & Center \$ ~ % * \$ * \$ * \$ * \$ * \$ Conditional Format as Neutral Calculational Conductional Styles nt \$ Number \$ * \$ * \$ * \$ * \$ * \$ * \$ * \$ * \$ * \$ *
Wrap Text General Conditional Normal Bad Merge & Center \$ - % * \$ & \$ & \$ & \$ & \$ & \$ & \$ & \$ & \$ & \$ &
Merge & Center * * * * Conditional Format as Formatting Neutral Calculatio nt r Number r Formatting Table Styles
nt ra Number ra Styles
6
168 > 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460 CK PERM=1
3 > 49168 [SVN, ACK] Seq=0 Ack=1 Win=8190 Len=0 iS=1436
168 > 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0
ent Hello
3 > 49168 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460 :P segment of a reassembled PDU]
3 > 49168 [PSH, ACK] Seq=1461 Ack=308 Win=35381
rver Hello, Certificate
168 > 443 [ACK] Seq=308 Ack=2921 Win=33028 Len=0
rver Key Exchange, Server Hello Done
100 > 443 [ACK] Seq=300 ACK=30U1 WIN=32348 Len=0
ndshake Message
3 > 49168 [ACK] Seq=3601 Ack=434 Win=35255 Len=0
ange Cipher Spec, Encrypted Handshake Message
168 > 443 [ACK] Seq=434 Ack=3652 Win=32297 Len=0
plication Data
3 > 49168 [ACK] Seq=3652 Ack=1111 Win=39848 Len=0
plication Data
crypted Alert
168 > 443 [FIN, ACK] Seq=1142 Ack=3819 Win=32130 n=0
3 > 49168 [ACK] Seq=3819 Ack=1142 Win=39817 Len=0
3 > 49168 [FIN, ACK] Seq=3819 Ack=1143 Win=39817 n=0
168 > 443 [ACK] Seq=1143 Ack=3820 Win=32130 Len=0

Figure 20. Successful Balance Check \$110 (CSV)

Appendix G Basic Transaction Data

	~♪	3			7		Rasic Transaction Table - Excel	
	×	Cut Home	N Page Lay	out Formulas	A	R		
	P 		Calibri	~ 11 ~ A	A = =	- 1%	PWrap Text General Mormal	Bad
Paste	4. 0	Format Painter	в <u>г</u> <u>и</u> -	- 🖓 - A			Merge & Center ~ \$ ~ % ? Conditional Format as Neutral	Calculation
	Clipt	board	2	Font	Ð	Alig	gnment rs Number rs	styles
G28		•	r × ×	**				
<u>.</u>	A	00	C	D	m	T	6	т -
-1 N		ime S	ource	Destination	Protocol	Length	Info	
,	•				T)	49165 > 443 [SYN] Seq=0 Win=32768 Len=0 MSS=1460	
~	F	0	92.100.0.7	200./1.1/.21	ç	20		
ω	Ν	0.103904 2	06.71.17.21	192.168.0.7	TCP	60	443 > 49165 [SYN, ACK] Seq=0 Ack=1 Win=8190 Len=0 MSS=1436	
4	ω	0.111887 1	92.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] Seq=1 Ack=1 Win=33028 Len=0	
Ś	4	0.381249 1	92.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello	
							443 > 49165 [PSH, ACK] Seq=1 Ack=308 Win=35381 Len=1460 [TCP	
6	ſ	0.41254 2	06./1.1/.21	192.168.0./	ICP	1514	And the final work of a reassembled PUUJ	
7	R	0 413755 2	06 71 17 21	192 168 0 7	TCP	1514	TCP segment of a reassembled PDLI	
~	7	0.414071 2	06.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certificate	
9	∞	0.414577 1	92.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] Seq=308 Ack=2921 Win=33028 Len=0	
10	9	0.42667 2	06.71.17.21	192.168.0.7	TLSv1.2	396	Server Key Exchange, Server Hello Done	
=	10	0.426977 1	92.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] Seq=308 Ack=3601 Win=32348 Len=0	
							Client Key Exchange, Change Cipher Spec, Encrypted Handshake	
12	11	0.905932 1	92.168.0.7	206.71.17.21	TLSv1.2	180	Message	
13	12	0.945464 2	06.71.17.21	192.168.0.7	TCP	60	443 > 49165 [ACK] Seq=3601 Ack=434 Win=35255 Len=0	
14	13	0.945491 2	06.71.17.21	192.168.0.7	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message	
15	14	0.98003 1	92.168.0.7	206.71.17.21	TLSv1.2	731	Application Data	
16	15	1.003492 2	06.71.17.21	192.168.0.7	TCP	60	443 > 49165 [ACK] Seq=3652 Ack=1111 Win=39848 Len=0	
17	16	1.506922 2	06.71.17.21	192.168.0.7	TLSv1.2	221	Application Data	
18	17	1.639561 1	92.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] Seq=1111 Ack=3819 Win=32130 Len=0	
19	18	1.710325 1	92.168.0.7	206.71.17.21	TLSv1.2	85	Encrypted Alert	
20	19	1.788997 2	06.71.17.21	192.168.0.7	TCP	60	443 > 49165 [ACK] Seq=3819 Ack=1142 Win=39817 Len=0	
21	20	1.801319 1	92.168.0.7	206.71.17.21	TCP	60	49165 > 443 [FIN, ACK] Seq=1142 Ack=3819 Win=32130 Len=0	
22	21	1.825801 2	06.71.17.21	192.168.0.7	TCP	60	443 > 49165 [FIN, ACK] Seq=3819 Ack=1143 Win=39817 Len=0	
23	22	1.826029 1	92.168.0.7	206.71.17.21	TCP	60	49165 > 443 [ACK] Seq=1143 Ack=3820 Win=32130 Len=0	
24								
25								
26								
27								
-	Ψ.	1 Balan	ce Check Inco	rrect PIN	+		-	····
Ready								

Figure 21. Basic Transaction Data (CSV)
Appendix H Packet Sequence Comparison by Packet Number

Table 7

Packet Sequence Comparison by Packet Number

No.	Source	Destination	Protocol	Length	Info
					49165 > 443 [SYN] Seq=0 Win=32768
1	192.168.0.7	206.71.17.21	ТСР	62	Len=0 MSS=1460 SACK_PERM=1
					49166 > 443 [SYN] Seq=0 Win=32768
1	192.168.0.7	206.71.17.21	ТСР	62	Len=0 MSS=1460 SACK_PERM=1
					49167 > 443 [SYN] Seq=0 Win=32768
1	192.168.0.7	206.71.17.21	ТСР	62	Len=0 MSS=1460 SACK_PERM=1
			_		49168 > 443 [SYN] Seq=0 Win=32768
1	192.168.0.7	206.71.17.21	ТСР	62	Len=0 MSS=1460 SACK_PERM=1
	400 400 0 7	206 74 47 24	TOD	62	49169 > 443 [SYN] Seq=0 Win=32768
1	192.168.0.7	206.71.17.21	TCP	62	Len=0 MSS=1460 SACK_PERM=1
1	102 169 0 7	206 71 17 21	TCD	62	491/0 > 443 [SYN] Seq=0 Win=32/68
T	192.108.0.7	200.71.17.21	TCP	62	Len=0 WISS=1460 SACK_PERM=1
					442 > 4016E [SVN ACK] Sog-0 Ack-1
2	206 71 17 21	192 168 0 7	тср	60	Win=8190 Len=0 MSS=1436
2	200.7 1.17.21	192.100.0.7		00	443 > 49166 [SVN ACK] Seq=0 Ack=1
2	206.71.17.21	192.168.0.7	ТСР	60	Win=8190 Len=0 MSS=1436
			-		443 > 49167 [SYN, ACK] Seg=0 Ack=1
2	206.71.17.21	192.168.0.7	ТСР	60	Win=8190 Len=0 MSS=1436
					443 > 49168 [SYN, ACK] Seq=0 Ack=1
2	206.71.17.21	192.168.0.7	ТСР	60	Win=8190 Len=0 MSS=1436
					443 > 49169 [SYN, ACK] Seq=0 Ack=1
2	206.71.17.21	192.168.0.7	ТСР	60	Win=8190 Len=0 MSS=1436
					443 > 49170 [SYN, ACK] Seq=0 Ack=1
2	206.71.17.21	192.168.0.7	ТСР	60	Win=8190 Len=0 MSS=1436
				60	49165 > 443 [ACK] Seq=1 Ack=1
3	192.168.0.7	206./1.17.21	ICP	60	Win=33028 Len=0
2	102 102 0 7	206 74 47 24	TCD	60	49166 > 443 [ACK] Seq=1 Ack=1
3	192.168.0./	206./1.1/.21	ICP	60	vvin=33028 Len=0

					49167 > 443 [ACK] Seg=1 Ack=1
3	192.168.0.7	206.71.17.21	ТСР	60	Win=33028 Len=0
					49168 > 443 [ACK] Seq=1 Ack=1
3	192.168.0.7	206.71.17.21	ТСР	60	Win=33028 Len=0
					49169 > 443 [ACK] Seq=1 Ack=1
3	192.168.0.7	206.71.17.21	ТСР	60	Win=33028 Len=0
					49170 > 443 [ACK] Seq=1 Ack=1
3	192.168.0.7	206.71.17.21	ТСР	60	Win=33028 Len=0
4	192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
4	192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
4	192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
4	192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
4	192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
4	192.168.0.7	206.71.17.21	TLSv1.2	361	Client Hello
					443 > 49165 [PSH, ACK] Seq=1
					Ack=308 Win=35381 Len=1460 [TCP
5	206.71.17.21	192.168.0.7	ТСР	1514	segment of a reassembled PDU]
					443 > 49166 [PSH, ACK] Seq=1
					Ack=308 Win=35381 Len=1460 [TCP
5	206.71.17.21	192.168.0.7	ТСР	1514	segment of a reassembled PDU]
					443 > 49167 [PSH, ACK] Seq=1
					Ack=308 Win=35381 Len=1460 [TCP
5	206.71.17.21	192.168.0.7	ТСР	1514	segment of a reassembled PDU]
					443 > 49168 [PSH, ACK] Seq=1
					Ack=308 Win=35381 Len=1460 [TCP
5	206.71.17.21	192.168.0.7	ТСР	1514	segment of a reassembled PDU]
					443 > 49169 [PSH, ACK] Seq=1
					Ack=308 Win=35381 Len=1460 [TCP
5	206.71.17.21	192.168.0.7	ТСР	1514	segment of a reassembled PDU]
					443 > 49170 [PSH, ACK] Seq=1
					Ack=308 Win=35381 Len=1460 [TCP
5	206.71.17.21	192.168.0.7	ТСР	1514	segment of a reassembled PDU]
					443 > 49165 [PSH, ACK] Seq=1461
					Ack=308 Win=35381 Len=1460 [TCP
6	206.71.17.21	192.168.0.7	ТСР	1514	segment of a reassembled PDU]

6	206.71.17.21	192.168.0.7	ТСР	1514	443 > 49166 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TCP segment of a reassembled PDU]
6	206.71.17.21	192.168.0.7	ТСР	1514	443 > 49167 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TCP segment of a reassembled PDU]
6	206.71.17.21	192.168.0.7	ТСР	1514	443 > 49168 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TCP segment of a reassembled PDU]
					443 > 49169 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TCP
6	206.71.17.21	192.168.0.7	ТСР	1514	segment of a reassembled PDU] 443 > 49170 [PSH, ACK] Seq=1461 Ack=308 Win=35381 Len=1460 [TCP
6	206.71.17.21	192.168.0.7	ТСР	1514	segment of a reassembled PDU]
7	206.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certificate
7	206.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certificate
7	206.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certificate
7	206.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certificate
7	206.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certificate
7	206.71.17.21	192.168.0.7	TLSv1.2	392	Server Hello, Certificate
					49165 > 443 [ACK] Seq=308
8	192.168.0.7	206.71.17.21	ТСР	60	Ack=2921 Win=33028 Len=0 Server Key Exchange, Server Hello
8	206.71.17.21	192.168.0.7	TLSv1.2	396	Done
					49167 > 443 [ACK] Seq=308
8	192.168.0.7	206.71.17.21	ТСР	60	Ack=2921 Win=33028 Len=0
8	192.168.0.7	206.71.17.21	ТСР	60	Ack=2921 Win=33028 Len=0
8	192 168 0 7	206 71 17 21	тср	60	49169 > 443 [ACK] Seq=308 Ack=2921 Win=33028 Len=0
0	192.100.0.7	200.71.17.21		00	A0170 > AA3 [ACK] Sec = 308
8	192.168.0.7	206.71.17.21	ТСР	60	Ack=2921 Win=33028 Len=0
					Server Key Exchange, Server Hello
9	206.71.17.21	192.168.0.7	TLSv1.2	396	Done

					49166 > 443 [ACK] Seg=308
9	192.168.0.7	206.71.17.21	ТСР	60	Ack=2921 Win=33028 Len=0
					Server Key Exchange, Server Hello
9	206.71.17.21	192.168.0.7	TLSv1.2	396	Done
					Server Key Exchange, Server Hello
9	206.71.17.21	192.168.0.7	TLSv1.2	396	Done
_					Server Key Exchange, Server Hello
9	206.71.17.21	192.168.0.7	TLSv1.2	396	Done
0	200 71 17 21	102 109 0 7		200	Server Key Exchange, Server Hello
9	206./1.1/.21	192.168.0.7	ILSV1.2	396	Done
10	102 169 0 7	206 71 17 21	тср	60	49105 > 443 [ACK] Seq=308
10	192.108.0.7	200.71.17.21	ICF	00	ACK-SOUT WIII-S2548 Lett-U
10	102 169 0 7	206 71 17 21		100	Client Key Exchange, Change Cipher
10	192.108.0.7	200.71.17.21	12371.2	180	AD1C7 > AD2 [ACK] Sec=208
10	102 169 0 7	206 71 17 21	тср	60	49167 > 443 [ACK] Seq=308
10	192.108.0.7	200.71.17.21	ICF	00	
10	102 169 0 7	206 71 17 21	тср	60	49108 > 443 [ACK] Seq=308
10	192.108.0.7	200.71.17.21	ICF	00	ACK = 5001 WH = 52548 Left = 0
10	102 168 0 7	206 71 17 21	тср	60	49109 > 443 [ACK] Seq=308 Ack=2601 Win=22248 Lon=0
10	192.108.0.7	200.71.17.21	ICF	00	ACK = 5001 Will = 52548 Left = 0
10	102 169 0 7	206 71 17 21	тср	60	49170 > 443 [ACK] Seq=308 Ack=2601 Win=22248 Lon=0
10	192.108.0.7	200.71.17.21	ICF	00	ACK-3001 WIII-32348 Lell-0
					Client Koy Exchange Change Cinher
11	192 168 0 7	206 71 17 21	TI Sv1 2	180	Spec Encrypted Handshake Message
	192.100.0.7	200.71.17.21	12341.2	100	443 > 49166 [ACK] Seg=3601
11	206 71 17 21	192 168 0 7	тср	60	Ack=434 Win=35255 Len=0
	200.71.17.21	192.100.0.7		00	Client Key Exchange Change Cinher
11	192,168,0,7	206.71.17.21	TLSv1.2	180	Spec. Encrypted Handshake Message
	19212001017	2000/ 202/ 222	1201212	100	Client Key Exchange Change Cinher
11	192,168,0,7	206.71.17.21	TLSv1.2	180	Spec. Encrypted Handshake Message
	192.100.0.7	200.71.17.21	12011.2	100	Client Key Exchange Change Cinher
11	192.168.0.7	206.71.17.21	TLSv1.2	180	Spec. Encrypted Handshake Message
					Client Key Exchange Change Cinher
11	192.168.0.7	206.71.17.21	TLSv1.2	180	Spec. Encrypted Handshake Message
_					, , , , , , , , , , , , , , , , , , , ,
					443 > 49165 [ACK] Seg=3601
12	206.71 17 21	192 168 0 7	тср	60	Ack=434 Win=35255 Len=0

					Change Cipher Spec, Encrypted
12	206.71.17.21	192.168.0.7	TLSv1.2	105	Handshake Message
					443 > 49167 [ACK] Seq=3601
12	206.71.17.21	192.168.0.7	ТСР	60	Ack=434 Win=35255 Len=0
					443 > 49168 [ACK] Seq=3601
12	206.71.17.21	192.168.0.7	ТСР	60	Ack=434 Win=35255 Len=0
					443 > 49169 [ACK] Seq=3601
12	206.71.17.21	192.168.0.7	ТСР	60	Ack=434 Win=35255 Len=0
					443 > 49170 [ACK] Seq=3601
12	206.71.17.21	192.168.0.7	ТСР	60	Ack=434 Win=35255 Len=0
					Change Cipher Spec, Encrypted
13	206.71.17.21	192.168.0.7	TLSv1.2	105	Handshake Message
13	192.168.0.7	206.71.17.21	TLSv1.2	731	Application Data
					Change Cipher Spec, Encrypted
13	206.71.17.21	192.168.0.7	TLSv1.2	105	Handshake Message
					Change Cipher Spec, Encrypted
13	206.71.17.21	192.168.0.7	TLSv1.2	105	Handshake Message
					Change Cipher Spec, Encrypted
13	206.71.17.21	192.168.0.7	TLSv1.2	105	Handshake Message
_				_	Change Cipher Spec, Encrypted
13	206.71.17.21	192.168.0.7	TLSv1.2	105	Handshake Message
14	192.168.0.7	206./1.1/.21	TLSv1.2	/31	Application Data
	200 74 47 24	400 400 0 7	TOD	60	443 > 49166 [ACK] Seq=3652
14	206./1.1/.21	192.168.0.7		60	Ack=1111 Win=39848 Len=0
14	192.168.0.7	206./1.1/.21	TLSV1.2	731	Application Data
				60	49168 > 443 [ACK] Seq=434
14	192.168.0.7	206.71.17.21	ТСР	60	Ack=3652 Win=32297 Len=0
				60	49169 > 443 [ACK] Seq=434
14	192.168.0.7	206.71.17.21	ТСР	60	Ack=3652 Win=32297 Len=0
14	192.168.0.7	206.71.17.21	TLSv1.2	731	Application Data
4 -	200 74 47 24	400 400 0 7	TOD	60	443 > 49165 [ACK] Seq=3652
15	206./1.1/.21	192.168.0.7		60	Ack=1111 Win=39848 Len=0
15	206./1.17.21	192.168.0.7	TLSv1.2	221	Application Data
4 -		400 400 0 -			443 > 49167 [ACK] Seq=3652
15	206.71.17.21	192.168.0.7	ТСР	60	Ack=1111 Win=39848 Len=0

15	192.168.0.7	206.71.17.21	TLSv1.2	731	Application Data
15	192.168.0.7	206.71.17.21	TLSv1.2	731	Application Data
					443 > 49170 [ACK] Seq=3652
15	206.71.17.21	192.168.0.7	ТСР	60	Ack=1111 Win=39848 Len=0
16	206.71.17.21	192.168.0.7	TLSv1.2	221	Application Data
					49166 > 443 [ACK] Seq=1111
16	192.168.0.7	206.71.17.21	ТСР	60	Ack=3819 Win=32130 Len=0
16	206.71.17.21	192.168.0.7	TLSv1.2	221	Application Data
					443 > 49168 [ACK] Seq=3652
16	206.71.17.21	192.168.0.7	ТСР	60	Ack=1111 Win=39848 Len=0
					443 > 49169 [ACK] Seq=3652
16	206.71.17.21	192.168.0.7	ТСР	60	Ack=1111 Win=39848 Len=0
16	206.71.17.21	192.168.0.7	TLSv1.2	221	Application Data
17	102 168 0 7	206 71 17 21	тср	60	49105 > 443 [ACK] Seq=1111 Ack=2810 [Min=22120] on=0
17	192.108.0.7	200.71.17.21		00	Encrypted Alert
1/	192.108.0.7	200./1.1/.21	ILSVI.Z	83	Encrypted Alert 401.7×442 [ACK] Sec = 1111
17	102 169 0 7	206 71 17 21	тср	60	49107 > 443 [ACK] Seq=1111
17	192.100.0.7	200.71.17.21		221	Ack-3019 WIII-32130 Lell-0
17	200.71.17.21	192.108.0.7	TLSV1.2	221	Application Data
17	200.71.17.21	192.108.0.7		221	
17	192.108.0.7	200.71.17.21	11311.2	65	Encrypted Alert
18	192.168.0.7	206.71.17.21	TLSv1.2	85	Encrypted Alert
					49166 > 443 [FIN, ACK] Seq=1142
18	192.168.0.7	206.71.17.21	ТСР	60	Ack=3819 Win=32130 Len=0
18	192.168.0.7	206.71.17.21	TLSv1.2	85	Encrypted Alert
18	192.168.0.7	206.71.17.21	TLSv1.2	85	Encrypted Alert
					49169 > 443 [ACK] Seq=1111
18	192.168.0.7	206.71.17.21	ТСР	60	Ack=3819 Win=32130 Len=0
					49170 > 443 [FIN, ACK] Seq=1142
18	192.168.0.7	206.71.17.21	ТСР	60	Ack=3819 Win=32130 Len=0
					443 > 49165 [ACK] Seq=3819
19	206./1.17.21	192.168.0.7	ICP	60	Ack=1142 Win=39817 Len=0
4.0	200 74 47 24	402 462 0 7	TOD	60	443 > 49166 [ACK] Seq=3819
19	206./1.1/.21	192.168.0.7	ICP	60	ACK=1142 WIN=39817 Len=0

19	206.71.17.21	192.168.0.7	ТСР	60	Ack=
19	192 168 0 7	206 71 17 21	тср	60	4910 Ack:
19	192.108.0.7	206 71 17 21	TI Sv1 2	85	Encr
15	192.100.0.7	200.71.17.21	11371.2	05	1/13
19	206.71.17.21	192.168.0.7	ТСР	60	Ack=
					491
20	192.168.0.7	206.71.17.21	ТСР	60	Ack=
20	206 71 17 21	192 168 0 7	тср	60	443 Ack-
20	200.71.17.21	192.100.0.7		00	/101/
20	192.168.0.7	206.71.17.21	ТСР	60	Ack=
					443
20	206.71.17.21	192.168.0.7	ТСР	60	Ack=
					491
20	192.168.0.7	206.71.17.21	ТСР	60	Ack=
~~				~ ~	443
20	206./1.1/.21	192.168.0.7	ICP	60	ACK:
					443
21	206.71.17.21	192.168.0.7	ТСР	60	Ack=
					491
21	192.168.0.7	206.71.17.21	ТСР	60	Ack=
					443
21	206.71.17.21	192.168.0.7	ТСР	60	Ack=
			_		443
21	206.71.17.21	192.168.0.7	ТСР	60	Ack=
24	200 74 47 24	402 460 0 7	TOD	60	443
21	206./1.1/.21	192.168.0.7	TCP	60	ACK=
21	192 168 0 7	206 71 17 21	тср	60	491 Ack
		200., 1.1, .21			7 1011
					491
22	192.168.0.7	206.71.17.21	ТСР	60	Ack=
					491
22	192.168.0.7	206.71.17.21	ТСР	60	Ack=

50	443 > 49167 [ACK] Seq=3819 Ack=1142 Win=39817 Len=0
	49168 > 443 [FIN, ACK] Seq=1142
0	Ack=3819 Win=32130 Len=0
5	Encrypted Alert
	443 > 49170 [ACK] Seg=3819
0	Ack=1142 Win=39817 Len=0
	49165 > 443 [FIN ACK] Seg=1142
0	Ack=3819 Win=32130 Len=0
-	443 > 49166 [FIN, ACK] Seg=3819
0	Ack=1143 Win=39817 Len=0
	49167 > 443 [FIN, ACK] Seg=1142
0	Ack=3819 Win=32130 Len=0
	443 > 49168 [ACK] Seq=3819
0	Ack=1142 Win=39817 Len=0
	49169 > 443 [FIN, ACK] Seq=1142
0	Ack=3819 Win=32130 Len=0
	443 > 49170 [FIN, ACK] Seq=3819
0	Ack=1143 Win=39817 Len=0
	443 > 49165 [FIN. ACK] Seq=3819
0	Ack=1143 Win=39817 Len=0
	49166 > 443 [ACK] Seg=1143
0	Ack=3820 Win=32130 Len=0
	443 > 49167 [FIN, ACK] Seq=3819
0	Ack=1143 Win=39817 Len=0
	443 > 49168 [FIN, ACK] Seq=3819
0	Ack=1143 Win=39817 Len=0
	443 > 49169 [ACK] Seq=3819
0	Ack=1142 Win=39817 Len=0
	49170 > 443 [ACK] Seq=1143
0	Ack=3820 Win=32130 Len=0
	49165 > 443 [ACK] Seq=1143
0	Ack=3820 Win=32130 Len=0
	49167 > 443 [ACK] Seq=1143
0	Ack=3820 Win=32130 Len=0

22	192.168.0.7	206.71.17.21	ТСР	60	49168 > 443 [ACK] Seq=1143 Ack=3820 Win=32130 Len=0
22	206.71.17.21	192.168.0.7	ТСР	60	443 > 49169 [FIN, ACK] Seq=3819 Ack=1143 Win=39817 Len=0
23	192.168.0.7	206.71.17.21	ТСР	60	49169 > 443 [ACK] Seq=1143 Ack=3820 Win=32130 Len=0